NPS ARCHIVE
1966
HIEBER, G.

THE
GEORGE WASHINGTON
UNIVERSITY

DATA PROCESSING TRAINING FOR
MARINE CORPS OFFICERS

by

MAJOR George A. Hieber, USMC

Thesis
H52726

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

DATA PROCESSING TRAINING FOR MARINE CORPS OFFICERS

By

George A. Hieber

Bachelor of Science

Florida Southern College, 1952

A Thesis Submitted to the School of Government and
Business Administration of The George Washington
University in Partial Fulfillment of
the Requirements for the Degree of
Master of Business Administration

April 30, 1966

Thesis directed by

John F. McCarthy, Jr., AB, ML

Associate Professor of Business Administration

NPS Archive
1966
Hieber, G.

Thesis
H52726

Library
U. S. Naval Postgraduate School
Monterey, California

PREFACE

Many books and articles generally recognize that management personnel other than those in the data processing installation need to be oriented towards data processing if effective utilization of these costly computers is to evolve. Present emphasis in the U. S. Marine Corps is on training of data processing personnel with limited orientation or training of other managerial personnel in the capabilities and limitations of computers.

The author carries an additional military occupational specialty as a Data Processing Officer and has been involved with Marine Corps data processing for a number of years. He found this lack of training of other officers a perplexing problem to him in the conduct of everyday tasks. It soon became evident to him that all officers dealing with him should have adequate training in data processing. This problem has been recognized by some Marine Corps officials, but to date a standard system for training of officers has not evolved.

This presentation explores the need for managerial training, present Marine Corps status on such training and an approach pursued by the Marine Corps Air Station, Cherry Point, North Carolina. A survey by a leading management consulting firm is discussed as a portion of this exploration. Detail lesson plans for the conduct of a semi-standardized course are developed.

I have sought to combine the guidance offered by my thesis director with my own personal experiences in the effort to develop the lesson plans. It is hoped that the plans presented will serve as an aid in the formulation of a Marine Corps-wide course.

As every student knows, a thesis comes to completion only through the help of many people who contribute their ideas, time and special skills. Here, I can only mention a few by name: Lieutenant Colonel Jack W. Harris for his work at Cherry Point; Virginia Levy for providing skillful assistance in editing and Marie Chamberlin for her excellent typing of the thesis though severly handicapped by my abominable handwriting. Finally I am indebted to my wife Bettye, for providing a climate in which I could work productively.

TABLE OF CONTENTS

LIST OF TRANSPARANCIES

CHAPTER I

NECESSITY FOR MANAGEMENT TRAINING

Computer Usage

A huge international industry has started growing up in just the last few years -- the 5 billion dollar world-wide computer business. It is far and away the fastest growing international industry with sales in industrial nations zooming at a rate of between 20 and 22 per cent a year. Presently there are about 10 billion dollars worth of computers installed around the world. About 75 per cent of the total is in the United States.[1]

This widespread and increasing use of computers promises marked changes in business management during the latter years of the twentieth century. We are presently on the threshold of the so called "second industrial revolution" which has been dramatically described as the "age of automation."[2]

The U. S. Marine Corps constitutes a portion of this computer market and is presently actively engaged in the increasing acquisition of computers. There are many factors to consider for the Marine Corps'

---

[1]"The $5 Billion World Market for Computers," Business Week (February 19, 1966), p. 110.

[2]Otis Lipstreu, "Organizational Implications of Automation," Journal of the Academy of Management (August, 1960), p. 119.

CHAPTER I

[illegible]

[illegible]

A huge international industry has [illegible] in just the last few years -- the [illegible] billion dollar [illegible] computer industry. It is [illegible] the fastest growing international industry [illegible] industrial nations growing at a rate of between [illegible] and [illegible] per cent a year. Presently there are some [illegible] million dollars worth of computers installed around the world. About [illegible] per cent of the total is in the United States.[1]

This widespread and increasing use of computers produces [illegible] changes in business management during the present state of the [illegible] economy. We are presently on the threshold of the so called "second industrial revolution" which has been [illegible] described as the "age of automation."[2]

The U. S. Marine Corps constitutes a portion of this computer market and is presently actively engaged in the [illegible] acquisition of computers. There are many [illegible] to consider for the [illegible]

[1] "Use of [illegible] Computers," [illegible] ([illegible]), p. [illegible].

[2] [illegible], "[illegible] Implications of Automation," Journal of the Academy of Management ([illegible]), p. [illegible].

effective utilization of its computers, however, this study will be confined to training of management personnel. The necessity for data processing training of Marine Corps officers will be examined with particular attention directed to the following questions. Is data processing training of officers desirable? If so, what rank levels should be trained? How extensive should the training be?

The information used to answer these questions will be both primary and secondary in nature. Opinions of civilian and governmental experts will be examined via textbooks, periodicals, reports and interviews. Particular emphasis will be placed on the results of a managerial course conducted at the Marine Corps Air Station, Cherry Point, N. C. The information will be sorted and integrated, and a proposed plan for data processing training by the U. S. Marine Corps should evolve.

McKinsey and Company Survey

In order to determine what American industry has thus far learned about organizing and managing computer systems for optimum results, McKinsey and Company undertook an intensive survey of 27 companies in 13 different industries ranging from heavy manufacturing to retail distribution. Predominantly these were large companies, leaders in their industries who had extensive computer systems experience.[3]

Early in the course of this survey it became clear that the 27 companies, considered in terms of the relative success of their computer

---

[3] John T. Garrity, Getting the Most Out of Your Computer, A Survey of Company Approaches and Results, Prepared by McKinsey and Co. (New York: McKinsey & Co., 1963), pp. 1-5.

system efforts, fell into two distinct groups. Nine companies referred to as the "above average" or "lead" companies had recovered the start up as well as the current operating costs of their computer installations. Eighteen others, the "average companies," were still a long way from covering current outlays much less recovering their original investment.[4] Almost without exception, the "above average" companies had made broad-scale use of computer-based systems. Most of their operating divisions had major applications installed and more were under way. Typically, they had set up computer applications in most major functions of the company, from production to marketing and distribution, not just in the accounting area. Unlike the average companies, whose computer systems were generally confined to routine record keeping activities, the lead companies had also put the computer to work on the crucial decisions of the business: in sales forecasting, in manpower and production scheduling, in inventory management.

This is not to say that the "above average" companies had not used computer systems on routine office and accounting functions. They had, and in so doing many have reduced costs significantly. But the experience of the lead companies shows that it is only as the computer is brought to bear on a broad front and, in particular, on the key problems of the business, that it begins to realize its full potential.[5]

The survey indicated that the major problem in getting a payout from the computer is not technical but managerial and organizational. In

---

[4] Ibid., p. 6.

[5] Ibid., p. 7.

none of the companies could less than satisfactory results be traced to equipment failure. All the survey companies had mastered most of the technical demands of computer systems. Furthermore, analyses of other factors, such as company size and type of industry, indicated that these factors might influence but did not determine the level of accomplishment.

Comparing patterns of management and organization of the computer systems effort, on the other hand, revealed a whole complex of significant differences between the two groups of companies. By and large these differences seem to stem from management's concept of the computer. In the average company, management had seen the computer simply as a highly advanced accounting machine for performing traditional functions faster and -- hopefully -- at lower cost. Or it had viewed the computer as a mysterious device that only the initiate can master, and thus had delegated all responsibility for the effort to technically trained subordinates.

In the above average companies, management viewed the computer very differently. It saw the computer systems effort as a major economic resource to be used in running the business. Further, the computer effort was held no more sacrosanct than any other new corporate activity and thus was subject to the same management processes.

This difference in concept is evident in all important aspects of the computer effort. First, it shows up in the quality of leadership that corporate executives provide. Next, it is reflected in the planning and control tools that management has built into the computer systems program. Third, it is apparent in the role operating management plays. Finally, it is reflected in the caliber of the computer systems staff.

Computer systems success is more heavily dependent on executive leadership than on any other factor. No company achieved above average results without the active participation of top management. And where corporate management in effect abdicated its responsibilities, the results were seldom outstanding.[6]

This difference in executive leadership manifests itself in many ways. In the above average companies, the corporate computer executive is most often just one level below the chief executive. In the typical average company, by contrast, this computer executive is more likely to be two, three, or even more levels down. Furthermore, in the average company, top executives devote far less time to the management of the computer systems effort.

But even more pronounced than these tangible signs of executive leadership is the marked difference in the atmosphere that surrounds the effort. In the average company, management is likely to take only a sporadic and superficial interest. As a result, some computer systems managers feel the effort has been hindered, not helped by company executives. In the above average company, by contrast, it is apparent that corporate management has set clear-cut objectives ensuring that the computer program is focused on the major problems of the business. Adequate resources have been marshaled to get the job done, and the human and organizational barriers to progress in this still-new field have been brought down. Finally, top management reviews and challenges the plans and programs of

---

[6] Ibid., pp. 12-13.

the computer group; monitors progress; and insists on significant, tangible benefits from its investment in computer systems.[7]

In keeping with its leadership role, corporate management in the above average company has required the development of tools for planning and controlling the computer systems effort. Typically, each computer application is approved only after a careful feasibility study that realistically weighs costs and risks against expected dollar payout. Before a computer project is launched, detailed project plans are drawn up, covering time and manpower requirements and setting interim goals. During the course of the project, periodic progress reports are made, measuring interim cost and accomplishment against plan. Finally, management requires and reviews an economic appraisal of completed projects to ensure that anticipations are realized.

In the typical lead company, corporate executives see to it that operating management takes a major responsibility for the end results produced by computer systems applications. In the average company, by contrast, the technical staff tends to dominate all the important computer systems decisions, leaving operating management in the role of a grudging, or at best indifferent, spectator.

In the above average companies, operating management plays a major role every step of the way. It helps decide which computer applications should be undertaken. It critically examines all application plans. It contributes manpower to each project team, not merely to ensure that operating personnel can administer the program once it is installed.

---

[7] Ibid., p. 15.

Finally, operating management takes major responsibility for the success of the application. Its attitude is summed up in the words of one lead company operating executive: "I'd no more play dead for the computer experts than I would for any other staff group. Of course EDP is complex -- but you can stay on top of it if you take the time to understand it."[8]

In summary the survey findings suggest that computer systems success is primarily influenced by top management's own attitude and approach. Thus by viewing the computer not merely as a machine but as a new resource to be employed, some companies have worked out new approaches to long standing problems often with dramatic results.[9]

Consequences of Training

It is apparent that superior results from automatic data processing require a basic change in approach. We must demonstrate a willingness to rethink problems of a business in terms of goals not in the streamlining of existing procedures. Automation requires managerial imagination rather than technical proficiency. Imaginative management in the new world of computers has paid off handsomely.

Instead of realizing this most managers have become intimidated by the computer's complexity and allowed the technicians to decide how the computers are to be utilized. To overcome this management must recognize the problem and educate first class managers in the potential of this "sleeping giant." Not only must data processing managers be

---

[8] Ibid., p. 17.

[9] John T. Garrity, "Top Management and Computer Profits," Harvard Business Review (July-August, 1963), p. 6.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

educated in automatic data processing but other managers must be shown the potential of automatic data processing.[10]

It is true that complete technical mastery of computer hardware is difficult to achieve. However, managers need not be electronic wizards, but should have some lesser degree of orientation in the capabilities and limitations of computers.[11]

Even this will involve a considerable amount of education if top management is to support data processing effectively and understand and control the problems and costs involved.[12]

The executive level should be trained and indoctrinated in the methods, procedures and capabilities of new systems before they are implemented. Failure to conduct pretraining can result in the lack of ability to diagnose problems in the early stages and failure of executives to properly use and understand the system.[13]

The problem of training managers will not be an easy task. Finding and keeping adequate numbers of trained personnel has historically been a problem and a limitation to computer systems development. In order to get a workable system, it is necessary to bring together a wide variety of

---

[10] John Diebold, "The Still Sleeping Giant," Harvard Business Review (September-October), 1964.

[11] William A. Gill, "Economic Considerations in the Use of Electronic Computers," An address before the American Management Association during a briefing session on computer economics (New York: June 29, 1962).

[12] Dick H. Brandon, "Editorial on Management Education," Computer and Automation, (December, 1965), p. 39.

[13] Hyman N. Laden and T. R. Gildersleeve, Systems Design for Computer Application (New York: John Wiley and Sons, Inc., 1963), p. 6.

technical and managerial skills to plan, develop, implement and manage the system.[14] Experience in computer systems is relatively short in numbers of years.

In the same years, the number of electronic data processing systems has grown phenominally. The demand for personnel has been high. Even untrained personnel with the capability of learning the skills have been in demand. Conversely the supply of skilled personnel is relatively inelastic. Educational institutions need time to set up programs. It takes time to interest sufficient numbers to enroll in special courses. And lastly, it takes time to learn and build experience in new skills.[15]

This continuing growth of electronic data systems foreshadows a continued shortage of adequately trained and experienced personnel in the near future. Technological advances are being investigated which may decrease the number of personnel required. Giving weight to the possible advances, it has been estimated that at least 318,000 skilled personnel will be required by 1970.[16]

The problem of finding adequate personnel is not the whole problem. This training of management in the aspects of computer systems will be costly and frustrating, but important consequences can follow. The achievement of high dollar return; intangible benefits from greater speed such as

---

[14] John A. Bekker, "Automation -- Its Impact on Management," Advanced Management (December, 1959), pp. 20-24.

[15] *Ibid*.

[16] Dick H. Brandon, "The Computer Personnel Revolution," Advanced Management (December, 1959), pp. 22-23.

improved customer service and reduction in manufacturing cycle time; and improved availability of operating information are the most important from the organization point of view. For the executive the training will hold the key to future advancement. Without it he is ill-equipped and at the mercy of the technicians.[17]

---

[17]Garrity, "Top Management and Computer Profits."

CHAPTER II

THE MARINE CORPS AND COMPUTERS

Present Status and Use in the Marine Corps

Despite its relatively small size and limited technological requirements, there are some 30 computer systems in the Marine Corps today and several more on order. The computers are engaged primarily in Supply Accounting, Personnel Accounting, Fiscal and Disbursing, and other logistic support programs. In support of these systems are about 1,000 enlisted, 300 civilians, and fewer than 100 officers.

Nine of the Marine Corps computers are in mobile configurations in support of the major Fleet Marine Force Commands, and one of these nine has been in operation at DaNang, South Vietnam, since a very few days after the Marine Units went ashore there.[1]

General Wallace M. Greene, Jr., Commandant of the Marine Corps, has on two occasions publicly expressed his views on automatic data processing in the Marine Corps as follows:

> 1. Automatic Data Processing (ADP) is being utilized in personnel, supply and logistics functions. We are also applying ADP to intelligence collection, analysis and dissemination. Anywhere ADP will improve our operations, we will utilize it. We are going to train as many people as possible in order to use these systems to the fullest extent.

---

[1]Lt. Col. Arthur T. Hill, "Marine Corps Position and Philosophy on Computer Education," a Report to the Seminar of Military Computer Educators and Computer Center Directors, June 14 to June 17, 1965 (West Point: By the author, 1965), p. 1.

I am trying to adapt all general officers and senior colonels into ADP and computer techniques. They must know these new systems of management to increase their proficiency in the field as these techniques are being adapted into weapon systems, high-speed data collection and other aspects.[2]

2. Use of computers in support of personnel, supply and logistics functions in the United States Marine Corps continues to expand at an accelerated rate.

Tactical Data, Intelligence Collection and Analysis, and Command and Control Systems are well down the road to becoming an effective reality.

Calendar year 1965 has seen the selection and first steps toward operating a random access, real-time computer and communications system in support of the total supply effort. This will reduce reaction time to supply needs from days to hours. A Marine Corps Reserve Data Services Center has been established this year at Kansas City, Mo.

Plans are completed in the automation of the Marine Corps Institute to provide this large correspondence school the facility for offering improved educational opportunities to Marines throughout the world.

During 1965, a master plan for the development of the Marine Corps Integrated Information System has been developed. This system will provide timely and meaningful information to appropriate decision-making and operating levels to permit monitoring and adjustment of current functional activities and analysis of probable future effects of available courses of action in decision making. The staffing and training of personnel to implement this master plan is now underway.

The Mobile Data Processing Platoon went ashore with the Marine Amphibious Force in South Vietnam in March.

The Corps recognizes that the age of the computer is here and that every administrative logistical or tactical decision in the future will be affected by its unerring ability to keep facts in order and to present them in the form and at the time needed for decision. In order to make effective use of this wonderful tool, we must replace all fear and apprehension with knowledge and understanding at all levels of command. Thus we recognize the need for

---

[2]General Wallace M. Greene, Jr., Commandant of the Marine Corps, "Defense Top Management Annual Review," Armed Forces Management, (November, 1964).

ever increasing quantities of education in Automatic Data Processing at an ever increasing level of quality.[3]

With these few words the Commandant has pretty well summed-up the Marine Corps operational use of ADP, and the tremendous requirement for increasing the depth and breadth of training in this rapidly expanding field. There is no segment of Marine Corps activity today which is not directly influenced by the use of electronic computers, whether through performance, operational control, or processing and reduction of data for command and management decisions. The Marine Corps, like her sister services, is faced with the challenge of absorbing this accelerating computer technology into the broader frameworks of leadership and management.

<u>Computer Education in the Armed Forces</u>

The Department of Defense, and more particularly the military services, make up the largest single computer user group in the world today; and while the systems employed are not necessarily unique, the military environment is. Consequently, it appears logical that the military services with their high turn-over of personnel and very fluid manpower situation will require a very extensive amount of computer education facilities. It is neither logical nor practical to depend on the vendors of hardware to continue to teach our personnel in the quantities and to the level of proficiency required.

---

[3] *Ibid*.

For generalized training in ADP concepts and ADP management, greater use might be made of joint facilities such as the recently inplemented Defense Computer Institute. The details of a particular system of management, or the specifics of a particular brand or configuration of hardware will continue to require in-house training.

The ever expanding complexity of military management systems, command and control systems and weapons systems, all computer oriented, demands that every junior officer have an understanding of ADP and what it can do for him, as well as what he can do with it. The service academies are already providing this training in varying degrees, but large gaps exist in the total picture and will continue to exist until ROTC and other programs have added this requirement, and until the individual services have provided in-house training for the residue.

The need for emphasis on ADP training in the middle and senior level service schools has been recognized, but the implementation is slow because the preponderance of personnel who are sufficiently experienced and qualified to present such programs are deeply involved in the development of bigger and better systems. By pooling efforts and sharing the schools and the expertise available to run them, the services are making headway; and within a very few years we may completely eliminate the magic and the mystery of ADP in the minds of the officers, regardless of rank or position.[4]

---

[4]Lt. Col. Arthur T. Hill, p. 2.

Computer Education in The Marine Corps

With the exception of several locally devised and very general orientation courses at major commands with computer facilities, computer training in the Marine Corps has been limited for the most part to those personnel necessary to staff the ADP operating forces, and has been provided by hardware vendors' courses and those courses offered at Army, Air Force, and Navy facilities.

Since computer technology antedates the college days and basic professional training days of the entire cadre of senior officers, there is an urgent and immediate need for extensive training at this level and a staff study is currently under development to establish a data systems course embracing both computer concepts and systems analysis at the Marine Corps Command and Staff College in Quantico. Ultimately it is intended that a Data Systems School be established to cover the full range of courses from systems analysis and design through computer programming and operations, tailored to the needs of the Corps in all pay grades and ranks. This school would then be the focal point from which instruction teams would be sent to present data systems courses in other locations as required.[5]

Present education of Marine Corps managers is divided into two classifications: General Officers and Other Officers. General Officers: it is difficult to stipulate the quantity and depth of orientation/training required for any general officer. However, since their duties are so diversified, their level of operation at the very top, and their time so

---

[5] *Ibid.*, p. 3.

limited, an orientation of capabilities, limitations, and exposure to methodology and users taking about a week's time would seem to be optimum. This is already packaged in a Department of Defense Computer Institute course. The Marine Corps has fully utilized its quota for this orientation course and as of January 1, 1966, had fourteen general officer graduates.

In addition, various Automatic Data Processing Equipment manufacturers offer seminar/orientation type courses for top management. Some are specifically tailored for general officers. The Marine Corps uses these facilities as the need arises and as of January 1, 1966, there were fifteen general officer graduates of these courses.[6]

An Automatic Data Processing Orientation Course is presently being conducted internally at Headquarters Marine Corps. This course taught by a Marine Officer is specially tailored to fit the training requirements at that headquarters. The course is open to general officers and two have attended as of January 1, 1966.[7]

Other Officers: it is impossible to construct a statistical summary of Automatic Data Processing training/orientation for all other officers since no collection of this data was ever required.

The courses offered by the Marine Corps to these officers are very limited in scope. Marine Corps Schools, Quantico, Virginia include

---

[6]Headquarters, Marine Corps, Data Systems Division, Report Submitted to the Office of Management Information, Department of the Navy, (Washington: 1966).

[7]Interview with Lt. Col. Arthur T. Hill, Executive Officer, Data Systems Division, Headquarters Marine Corps, February 17, 1966.

survey type courses in Automatic Data Processing as a portion of the curriculum at the Amphibious Warfare School and the Command and Staff College. Not only are these courses limited in scope but they are limited in input. The annual quotas are: Amphibious Warfare School, 380 Captains/Majors; Command and Staff College, 106 Majors/Lieutenant Colonels. This total is insignificant when compared to total officer strength.

The Marine Corps Institute offers two, six lesson, twenty-nine hour study courses primarily designed for enlisted Marines desirous of entering the Automatic Data Processing field. While these courses are very rudimentary they are a good primer for the field and a means of initiating any novice. They are available to all military personnel.

As in the case of the general officers there is the orientation course conducted at Headquarters Marine Corps. Officers of that headquarters attending are generally those requiring immediate knowledge of Automatic Data Processing for use in their jobs.

Much thought has been given by the Marine Corps to the training of all officers in Automatic Data Processing. Many officers now entering the Marine Corps have already received much formal training in the field, i.e., the service academy graduates. However, not all of the new officers are academy graduates nor has their previous education included Automatic Data Processing. As a minimum the Marine Corps recommends that Automatic Data Processing education equivalent to that given at the service academies be given to all officers at the Basic School, Marine Corps Schools.[8]

---

[8] Report Submitted to the Office of Management Information.

Those officers who have finished Basic School and have not been exposed to any formal Automatic Data Processing education could receive their education via Marine Corps Institute, Amphibious Warfare School, Command and Staff College, manufacturers courses, or local command orientation.

The most fruitful area would seem to be in the area of local command orientation. A centrally designed course curriculum which would allow modification for specific local requirements should optimize training. This seems to be the present thinking at Headquarters Marine Corps. It is planned to export their internal orientation course to the Marine Corps Recruit Depot, San Diego, California. The course will be conducted by the present instructor since no detailed lesson plans are available for use by the local command. Concern for the need of a semi-standardized course with detailed lesson plans has been expressed by Marine officials. Once these are available the Automatic Data Processing education of Marine officers could proceed at full speed, thus assuring the optimum utilization of the Marine Corps computer systems.[9]

---

[9] Interview with Lt. Col. Arthur T. Hill.

CHAPTER III

THE CHERRY POINT APPROACH AS RELATED TO MARINE CORPS NEEDS

Background

From 1962 to 1964 the author was Systems Development Officer and Deputy Data Processing Officer, Marine Corps Air Station, Cherry Point, North Carolina. During 1964 Cherry Point was in the process of preparing for the installation of a large scale real-time computer. Cherry Point had employed a small scale card computer for three years. This small card system was primarily used in the supply, financial, and personnel areas. Being a small card system only a few management personnel in each functional area were involved and there was very little interface between the areas. These managers had few interrelationships.

The new system will change this completely. The Bureau of Naval Weapons for a number of years had been working on design of an integrated management information system for its seven Industrial Naval Air Stations which included Cherry Point. An Industrial Naval Air Station is one which has aircraft overhaul and repair facilities. This system was being centrally designed by the Bureau of Naval Weapons with the assistance of systems analysts from the seven Industrial Naval Air Stations. In concept it would cover all aspects of material control, financial control, workload and production control, and personnel accounting at the stations. No

longer would each area be separate as in the past but there would be a dovetailing and crossing of functional lines. A broad spectrum of management personnel in every area will become involved with the new system. Many management positions will change and be replaced with newer and more complex functions.

Design of the integrated management information system progressed satisfactorily. In August, 1964 the computer system, an RCA 3301 which included a 120 K high speed memory, 10 magnetic tape stations, a 1.3 million character magnetic drum, random access computer equipment with total storage capacity of more than 680 million characters, remote inquiry equipment and the normal input-output devices was selected by the Bureau of Naval Weapons. The Command realized that its management personnel were ill-prepared for the advent of the new computer system. In September, 1964, a plan evolved to train these personnel.[1]

Plan of Endeavor

The Station Management Information Systems Working Group was given the responsibility of standardizing a course of instruction for Cherry Point management. This management Information Working Group was established in 1962 with the Management Engineer as Chairman and the author as Vice Chairman. The group was charged with the responsibility of adapting and implementing the Bureau of Naval Weapons integrated management information system at Cherry Point. It was only natural that the task of training management should fall to this group.

---

[1] Air Bases Order 1500.8, Marine Corps Air Bases, Eastern Area, Cherry Point, North Carolina, 14 October 1964.

The computer system was to be installed in two phases. The first phase called for installation of the system, less remotes and random access equipment in January, 1965, with the second phase following in six months. Time and availability of trained personnel were critical factors, therefore the station data processing personnel could not be called on to act as instructors for the entire course. The Management Information Systems Working Group decided upon a unique plan to solve the instructor problem. The instructor load was spread out throughout all departments to minimize interference with regular duties. A pool of twelve instructors was established. Department quotas were: Overhaul and Repair, three; Supply, Data Processing and Public Works, two each; Industrial Relations, Comptroller and Management Engineer, one each. It was not required that the instructors have had some previous knowledge of data processing, only that they be capable individuals in their field.[2]

The instructor training conducted by the Management Engineer and the author was devoted entirely to the elementary aspects of data processing with particular emphasis on the RCA 3301 computer system. The training required eighty hours and was conducted during the period October 19 to October 30, 1964.

Development of the syllabus proceeded post-haste. Detailed lesson plans with appropriate training aids were developed. The instructors were divided into six teams of two each. The teams were each responsible for five hours of the total of thirty hours of instruction. It was envisioned

---

[2] *Ibid*.

that approximately three hundred seventy-five management personnel in groups of twenty-five each would receive the training.

The course syllabus was as follows:

A. Introduction

B. Processors

C. Storage Devices

D. Input/Output

E. Stored Program Concepts and Systems Design

F. Programming

G. Hypothetical Computer

H. 3301 Phase I

I. 3301 Phase II

J. Software

K. Material Application

L. Workload Application

M. Financial Application.[3]

Purpose and Results

The purpose of this plan was threefold:

1. To orient management personnel about data processing, the RCA 3301 computer system, and the integrated information system.

2. To utilize the instructors as a base for contact by personnel in each department. This it was believed, would serve to alleviate problems as they occurred during implementation.

---

[3]Air Station Bulletin 1500, Marine Corps Air Station, Cherry Point, North Carolina, October 29, 1964.

3. To create a feeling of commonality between the Data Processing Department and its customers.

The first group to attend was composed of the Commanding General and his primary staff officers. The author attended all class sessions for this group. It was noted that the instructors performed very well and that the initial evaluation of the course by these managers was favorable.

To date one hundred thirty-nine military and two hundred nineteen civilians for a total three hundred fifty-eight management personnel have attended this course.

Critique sheets on the course were filled out by the attendees. An analysis of these questionnaires revealed that:

1. 93 per cent thought the course was logically organized.

2. 76 per cent felt that the time devoted to each part of the course was adequate.

3. 69 per cent thought the time was fully and effectively utilized.

4. 94 per cent felt the course was well presented.

5. 85 per cent thought the information received would help them better accomplish their job.

6. 94 per cent described their overall reaction to the course as satisfactory with 69 per cent rating it very good to excellent.[4]

Some minor revisions to the course have been made since implementation. The course has been shortened in length but the organization of it has remained intact.

---

[4]Letter from Mr. William W. Waters, Deputy Data Processing Officer, Marine Corps Air Station, Cherry Point, North Carolina, February 21, 1966.

The implementation of the new system has progressed very smoothly. In fact Cherry Point was the last Air Station to receive the new computer system and the first to complete conversion and release the old computer. The training program has been a factor in this highly satisfactory performance. Knowledge of this program and its apparent benefits has spread to other Air Stations. At present the course is being utilized by two other Industrial Air Stations.[5] Thus the success of the Cherry Point approach seems assured.

Factors Requiring Recognition

The Cherry Point experience seems to bear out that management education is of vital importance to the continuing success of a data processing installation and to the effective use of technology. Major reasons for such education are:

1. Management must understand the problems to cope with them and provide effective solutions.

2. Management must recognize the potential of data processing.

3. The increasing commonality of the data base require interdepartmental integration, which in turn requires management direction.

4. Executives must be aware of the lead time involved in data processing to provide organization-wide long-range planning.

5. Management must understand the need to spend money for research, equipment, planning, standards and documentation.

---

[5] Interview with Mr. William W. Waters, Deputy Data Processing Officer, Marine Corps Air Station, Cherry Point, North Carolina, February 10, 1966.

The implementation of the new system has progressed very smoothly. In fact Cherry Point was the lead Air Station to receive the new computer system and the first to complete conversion and release the old computer. The training program has been a factor in this highly satisfactory performance. Knowledge of this program and its apparent benefits has spread to other Air Stations. At present the concept is being utilized by the Naval Industrial Air Stations.[3] Thus the success of the Cherry Point approach seems assured.

Factors Requiring Recognition

The Cherry Point experience tends to bear out that management education is of vital importance to the continuing success of a data processing installation and to the effective use of technology. Major reasons for such education are:

1. Management must understand the problems to cope with them and provide effective solutions.

2. Management must recognize the potential of data processing.

3. The increasing complexity of the data used permits inter-departmental integration, which in turn requires management direction.

4. Executives must be aware of the lead time involved in data processing to provide organization-wide long-range planning.

5. Management must understand the need to spend money for research, equipment, planning, training and documentation.

---

[3]Interview with Mr. William H. Wright, Deputy Data Processing Officer, Marine Corps Air Station, Cherry Point, North Carolina, February 18, 1966.

6. Management must evaluate and control the installation performance.

7. Management must avoid making unreasonable demands on the resources of the installation.[6]

The objectives of a good management training course must recognize these factors. Such a course must mix hardware fundamentals with an evaluation of the impact of data processing.

Composition of Syllabus

Two outlines besides the Cherry Point approach for a successful course are as follows:

A. The Nature of Automation
   1. Types of Automation
   2. General impact

B. Information Processing Systems
   1. Types of systems
   2. Stored programming concepts

C. Computer Classification
   1. By type
   2. By size
   3. By function

D. Components of Computers
   1. Input
   2. Output

---

[6]Dick H. Brandon, "Editorial on Management Education, " Computers and Automation, December, 1965, p. 39.

3. File storage
4. Main storage

E. Communications with Computers
1. Data representation
2. Language levels
3. Personnel functions
4. Software

F. Planning Steps -- Feasibility
1. The feasibility study
2. Equipment selection
3. Contract negotiation

G. Planning Steps -- Prerequisites
1. Personnel selection
2. Personnel training
3. Organizational structure
4. Standard development
5. Scheduling and budgeting
6. Site requirements

H. Planning Steps -- Implementation
1. Requirements
2. Systems design
3. Programming

I. Planning Steps -- Installation
1. Conversion
2. Systems Testing

3. Take over

4. Audit

J. The Role of Management

1. Data processing management

2. User management

3. Top management.[7]

II. A. Introduction

B. Marine Corps Data Systems Organization

1. HQMC

2. Posts and stations

3. FMF Units

4. Budget

5. Personnel

6. Limitations

7. Plans for the future

C. Punched-card machines -- Concepts and Systems

1. Early Developments (Babbage)

2. Basic Concepts

3. Limitations

4. Common Equipment Types and Purposes

D. Electronic Computers -- Basic Concepts

1. Types and Purposes

2. History of development

---

[7] *Ibid*.

3. Functional elements
   Input/Memory/Arithmetic and Logic/Control/Output

4. Communications with computers

E. Electronic Computer Components

1. Input/Output/CPU/Mass Memory/specials

2. The concepts of random access and real time

F. EDP Software Logic and Concepts

1. Machine Language Principles

2. Binary and Octal Arithmetic

3. Program Languages

   (a) Machine/Pseudo/Symbolic/Automatic

G. Systems Analysis and Programming Concepts

1. Applications Study

2. Feasibility

3. General Flow Diagrams

4. Detailed Flow Charts

5. Coding

6. Debug/prove

7. Implement

H. USMC Personnel Services Support

1. Present programs and concepts, Regular and Reserve

2. Future plans.

I. USMC Material Services Support

1. Present Programs, Supply, Fiscal, Disbursing and Logistics support

2. Future Plans, including MUMMS, CAREPAY, etc.

J. Future Considerations in ADP for USMC

1. Hardware

2. Systems Concepts

3. Command and Control

4. Integrated Management Information

K. Closing Summary.[8]

Both these outlines appear too detailed and lengthy for a management orientation course. The Cherry Point experiment indicated that it is best to make an orientation course as short and concise as possible to insure success. That course was shortened by twenty-five per cent in order to overcome certain elements of management who resisted any type of training. It was decided that managers who required additional information could take specialized courses. This action did not weaken the original purposes of the plan but the net effect was to strengthen them.[9]

Based upon the foregoing a variation of the Cherry Point course would seem most appropriate for a general orientation course for Marine Corps management personnel who have not had previous Automatic Data Processing training. Additional training required by selected managers could be provided on an individual basis as determined by Marine Corps authorities.

Since the type of data processing equipment and the utilization

---

[8]Syllabus, Automatic Data Processing Orientation Course, Headquarters Marine Corps, Washington, D.C., February, 1966.

[9]Letter from William W. Waters.

of it differs from installation to installation, a completely standardized course does not seem feasible. In all probability the course should be semi-standardized and comprised of two main sections:

1. A standard section on data processing in general with detailed lesson plans and training aids. This section should be centrally developed.

2. A non-standard section on hardware, software and systems application utilized by local installations. The section should be developed by each command and tailored to fit its needs.

CHAPTER IV

DESIGN OF A COURSE FOR MANAGERS

Contents of Course

The standard section of the course is presented in this chapter. It is based upon and is a modification of Introduction to Automatic Data Processing and UDAPS for INAS.[1]

The detailed lesson plans follow:

[1] Jack W. Harris and George A. Hieber, Introduction to Automatic Data Processing and UDAPS for INAS (Cherry Point, North Carolina: By the authors, 1964).

TABLE OF CONTENTS

A -- INTRODUCTION

Lesson Plan

COURSE:	AUTOMATIC DATA PROCESSING
SESSION TITLE:	INTRODUCTION
OBJECTIVE:	1. To build a base 2. Remove the mystery
TIME:	50 minutes
TRAINING AIDS REQUIRED:	Overhead projector and one transparency Pointer Film: "Introduction to Automatic Data Processing," "MN 8969A (Available from Film Library, MCAS, Cherry Point, N.C.) Projector Glossary of terms

Introduction

We are going to talk today and for a few days to come about Automatic Data Processing. During this morning's introductory remarks, I am going to outline for you the overall course objective, touch on the history and impact of data processing, go over a glossary of terms with you, and show you an introductory film about data processing. OUR OBJECTIVE IS A THREE FOLD ONE:

FIRST: We want to build a base which will give you a clear understanding of the potential of a Computer System which will enable you to meet not only the demands of your Automatic Data Processing System, but will also enable you to build on, or subtract from the system as your work load demands. In other words, we hope to make this program so well understood that it can be adapted to any condition or situation which may arise. In order to build a base, we are going to have to get basic. So, you will spend the first few hours on the basics of data processing systems. We have a small hypothetical computer here in the classroom -- we will explain how it works and you will have an opportunity to program on a computer and see for yourselves how it is done. You will find this interesting because here is an opportunity for you to understand the workings of the machine and when you understand the machine, it is simple.

SECOND: We want to remove the aura of mystery surrounding Data Processing. What is a data processing system? A data processing system as we know it is made up of five basic components -- SOME KIND OF INPUT -- SOME SORT OF CONTROL -- SOME SORT OF PROCESSOR -- THE ABILITY TO PERFORM ARITHMETIC AND LOGICAL OPERATIONS -- AND AN OUTPUT. Let me show you a

picture of the first computer. (SHOW TRANSPARENCY #1) You will notice that the first computer was a person. This person has the required five basic parts: Input, control, processor, ability to do arithmetic and logical operations and output.

You will recognize this form of input -- the faithful old inbox, the means by which information enters the processor. Here is the CONTROL mechanism or the BRAIN which performs the logic and arithmetic -- getting assistance on the arithmetic from the adder or adding machine. The PROCESSOR is controlled by procedures or a set of instructions which tell her what to do based on previous experience. You will notice the telephone which may represent a remote input and also possibly a means of output, working storage is here in the form of a scratch pad, and here is information necessary to the processor in the form of files and other records, and finally we have an output, here in the outbox. (OFF TRANSPARENCY #1)

An electronic computer or processor will provide the same service and perform these functions in a basically similar manner, but without the handicap of emotion, or forgetfulness or any of the human frailties and with speed and accuracy the human computer could never hope to attain.

AND THE THIRD PORTION OF OUR OBJECTIVE is to help you to become oriented to the (substitute name of organization's computer) specifically. In the past you may have had computer experience -- but there are many different types available. You may have worked with IBM, RCA or Honeywell or some other and found it quite good. Whatever your past experience has been, you will find that the (substitute organization's computer) opens new horizons.

So, our objectives are to build a base, remove the mystery about computers, and familiarize you with the (substitute name of organization's computer). From the schedule you have received you know that we anticipate its taking approximately 18 hours to achieve this three-fold objective. We hope you will attend the course in its entirety, since you will find it difficult to maintain continuity if you miss sessions. We know your time is valuable, but we believe it will be more valuable after this course. Your being asked to attend so many meetings reminds me of a definition which you will not find on your glossary of terms. The definition of the difference between a drunkard and an alcoholic is that an alcoholic has to attend meetings. You too are being asked to attend meetings, but there the similarity ends. After these meetings, the difference between you and someone who has not been exposed will be that you will know (name of organization's computer).

So what? Why should a manager be concerned about ADP? What do we have automatic data processing personnel for? The reason is that ADP is becoming too important to leave to the automatic data processing people.

Why all this sudden emphasis on data processing. Data processing isn't something new. It's something we have had with us for years. Why suddenly the emphasis? BECAUSE OF THE INCREASED CAPABILITY MANAGEMENT HAS AS A RESULT OF COMPUTERS. They can give you the information you need to make decisions based on better, more timely facts. You can take actions in seconds which formerly took hours or even days. You can use this machine to forecast and predict the result of your decisions before you make them.

One prime example of this is the use of computer in predicting election results. Those who looked at their TV screens in the Nov. 1964 Presidential Election saw the computer being used as a management information system. NBC called this "Operation Ballot." What NBC and RCA did was make a mathematical model of the voting habits and history of the country. They fed the data into the system from past elections and polls and they used that to predict the outcome of the election based on returns from bellwether precincts. In other words, the network selected a sample represented by 3 thousand precincts from which they determined the outcome of the election. In general, NBC's model incorporated four classes of data. First the vote itself. Using highly sophisticated statistical techniques, it was possible to notice trends developing as initial votes were received. Projection of these trends produced an anticipated total. Secondly, the model put particular emphasis on the so-called bellwether precincts which historically had been good indicators of the voting trends in their respective geographic areas. The model also encompassed dozens of demographic factors for each county. These factors included the known voting effects of age, sex, income, registration, race, religion, etc. Finally, the model incorporated an assortment of additional factors which included results of public opinion polls and vast quantities of historical data. The voting behavior of all 3,000 precincts had been carefully charted for many years, and was "factored" into the model as required. What was the man-machine relationship in this operation? In essence, the vast computer complex provided an extension of its human operators'

intellect. Before the first vote was received on election night, the technical experts made many assumptions. As initial votes were received and projected, these experts used computer projections to refine their assumptions accordingly. This technique is called simulation -- that is the ability to make predictions based on probability. Thus, the computer provided a calculating tool for analyzing and projecting the effect of decisions.

Most of us who saw this were not likely to grasp the ultimate significance of what we saw. True, we were aware of the political implications of computer projections, and in a general way we appreciated the technological advances involved, but what is the true potential of such advanced communications? The answer is simple. We saw a sample of a most advanced management information system -- a system of almost startling possibilities for us as managers, in terms of new capability, speed and decision-making assistance.

Computers of the future will inevitably introduce changes in the way we work, in the way we learn and even in the way we provide for our armed defense.

Data processing as we now know offers many advantages. Primarily we are able to process data while it is timely, with almost unbelievable speed and accuracy -- if we understand how to do it.

One of the necessary things we must do in learning about data processing is to learn the language and I assure you that the computer has a language all its own. Some of the terms on the glossary you have been given will be used in the film we are showing this morning. Others you will hear used throughout the course.

We will not go over all the definitions now but I would like to call your attention to a few of them which you will need immediately. Please keep the glossary with you when in class so that you may use it for a ready reference. For example, turn to page 2 of the handout, and notice "Card Punch." This is the most common type of device for getting information or data into the computer.

1. Oldest form, used first over a hundred years ago.
2. Herman Hollerith developed it.
3. Can be read electro-mechanically.
4. Has 80 columns (the one we use).
5. Disadvantage:
   a. Limited number of characters is low because of the size of the holes punched in it.
   b. Bulky, hard to store, can't be mailed easily.
   c. Con't be altered or erased -- a one time record.
   d. Hard to transmit more than a few per minute over telegraph lines.

Sometimes there are times when the operator of the system needs to get in and stop a process, change something that is in error or his procedures call for an interrupt. The console is the operator's control over the processor. This is how he tells the processor what to do. Look at the bottom of page 2 of the handout for "console."

Another is "Magnetic Tape" --

1. Coated with iron oxide.

2. Densities of from 100 to 1500 characters per inch. An 80 column card can be reproduced on tape in .053 inches of tape.

3. Can be read at rates up to 200 inches a second. Compare this with the high speed card reader that will read 1470 cards a minute. This is getting fast, but not nearly as fast as the processor.

Another item is "Punched paper tape" -- turn to page 8.

1. Used the same as cards, somewhat easier to use and has been gaining wide acceptance.

2. We see it used daily in cash registers, transactors, long distance direct-dialing, etc.

3. It too is limited, however, as is the card number of characters due to hole size.

4. Can be mailed, but is easy to break and tear. Generally, there are four basic means of getting information or data into the system as we have seen.

a. Cards

b. Console

c. Magnetic tape

d. Paper tape

You will be hearing real-time, on-line, off-line and many other terms that have special meanings when used in connection with automatic data processing systems. Real-time refers to one type of ADPS. See the definition of "real-time" on page 8.

1. Basically a real-time system obtains data about operations while they are occurring, processes the data, and furnishes results quickly

enough to be useful for controlling the outcome of a process.

2. An example of a real-system is the airline ticket desk.

We ask for a reservation. The salesperson selects the proper punched-card, inserts it into his remote station and then keys in additional information such as number of seats, date wanted, etc. The answer comes back to us instantly.

The thing that I want to get across here is that a real-time system is one with its connections directly into the processor so that results are obtained in time to permit effective control action to be taken.

The off-line system is when the computer operates independently of the actual inputs, i.e., there is a time lag between the input and the resulting output.

(GO OVER ANY OTHER TERMS NECESSARY)

While those are fresh in your minds, let me introduce data processing to you through the medium of film.

(SHOW FILM)

SUMMARY

We have seen a film on automatic data processing systems, and we saw that it consists of the 5 P's -- Processor, Peripheral Equipment, Procedures, Programs, People.

1. Processor, the memory or brain

a. This is the difference between a data processing system and an automatic data processing system.

b. A processor has the means to do arithmetic and logical operations automatically.

2. <u>Peripheral Equipment</u>

   a. Input/output devices

   b. Printers

3. <u>Procedures</u>

   a. To tell, what is needed.

   b. To tell, when it is needed.

   c. To tell, where to obtain it.

   d. To tell, how to use it.

4. <u>Programs</u>

   a. When we think of the machine, we use the term programs rather than procedures.

   b. Programs are the routines for the processor. They tell it what to do.

5. <u>People</u>

   a. We need people to operate the equipment.

   b. We need people to maintain the equipment.

   c. We need people to analyze and set up procedures.

   d. We need people to provide input data.

   e. We need people to utilize reports and review results.

   f. We need people to supervise the entire operation.

Gentlemen, in this session we have outlined the objectives of the ADP Course, and explained why you are here.

We have pointed out some of the capabilities of modern data processing systems, and have discussed a few of the terms we will be working with.

We have seen a film introducing Data Processing, and mentioned the five P's -- each of which will be covered in detail as we go along. We have mentioned about the four input devices to a computer. With this background, we will next talk about the computer itself.

ARE THERE ANY QUESTIONS?

HANDOUT

Computer Language

Absolute Address -- An address expressed in machine language.

Access Time -- The time it takes a computer to locate data or an instruction word in its memory or storage section, and transfer it to its arithmetic unit where the required computations are performed. Also, the time it takes to transfer information which has been operated on, from the arithmetic unit to the location in memory where the information is to be stored. If you and your sliderule were functioning as a human computer, access time would be comparable to the time it took you to locate a figure on a log sheet and set it on your sliderule.

Accumulator -- A device which stores the results of arithmetic operations.

Address -- A label that identifies for the computer a specific location in its memory where certain information is stored. Serves the computer in much the same way that index tabs on your filing system serve you.

Arithmetic Unit -- The part of a computer that performs arithmetic and logical operations. In the case of you and your desk calculator, the latter is comparable to the computer's arithmetic unit.

Assemble -- To integrate subroutines and routines into a main program.

Binary Coded Decimal Representation (BCD) -- A system of representing decimal numbers. Each decimal digit is represented by a combination of four binary digits (bits) as follows:

Binary Code		Decimal Code
0000	=	0
0001	=	1
0010	=	2
0011	=	3
0100	=	4
0101	=	5
0110	=	6
0111	=	7
1000	=	8
1001	=	9
0001 000	=	10

Binary Digit (Bit) -- In the binary numbering system, only two marks (0 and 1) are used. Each of these marks is called a binary digit. Let's take the decimal number 296 and convert it to a binary number. It becomes 100101000, and we say that it is made up of nine binary digits or bits.

<u>Block</u> -- A group of words considered as a unit.

<u>Branch</u> -- See "Jump."

<u>Buffer</u> -- A device for compensating for differences in speed between two devices to permit them to operate together.

<u>Bulk Storage</u> -- Large-volume storage used to supplement the high-speed storage; may be addressable, as with disks and drums, or nonaddressable, as with magnetic tapes. Also called "secondary" and "external storage."

<u>Card Punch</u> -- A device for punching data in cards. Examples are simple hand punches, keyboard print-punches, paper-tape-to-card converter punches, and high-speed punches for magnetic-tape-to-card conversion, or for direct output from the processor.

<u>Check</u> -- Process of testing: (a) existence of certain prescribed conditions within the computer; (b) correctness of machine operation; or (c) correctness of results. Checks may either be programmed or made automatically by the computer.

<u>COBOL</u> -- Common Business Oriented Language; an English-like programming language designed primarily for business-type applications and implemented for use with many different data processors.

<u>Code</u> -- A system of characters and rules for representing information in a language that can be understood and handled by the computer.

<u>Collate</u> -- To produce a single sequence of items, ordered according to some rule, from two or more similarly ordered sequences. The final sequence need not contain all of the data available in the original sets. If, for example, two sets of items are being matched, items that do not match may be discarded.

<u>Computer</u> -- Any device capable of accepting data, applying prescribed processes to them, and supplying the results of these processes. The word "computer" usually refers to an internally-stored-program data processor; the term "processor" is preferable for business applications.

<u>Console</u> -- Equipment that provides for manual intervention and for monitoring processor operations.

<u>Control Unit</u> -- The section of a computer that controls all information transfers and arithmetic operations in the computer. In most computers, it also controls the sequence of operations, and initiates the proper commands to the computer circuits after decoding an instruction.

Converter -- a device for transferring data from one storage medium to another -- for example, from punched cards to magnetic tape.

Data -- Figures, words, or charts that refer to or describe some situation.

Data Density -- the number of characters that can be stored per unit of length, area, or volume. Specifically, for magnetic tape, the number of bits in one row per inch of tape where one bit in each row across the tape makes up a frame representing one character.

Data Origination -- the steps used for obtaining data at the points where events occur. The operations may be manual or mechanical and the data may be obtained in a form that requires conversion or that are directly usable for further processing.

Data Processing -- Rearrangement and refinement of data into a form suitable for further use; often involves file processing to update files for transactions that occur.

Debug -- to test a program by running it with test, simulated, or live data on a processor to find whether it works properly, and, if mistakes are revealed either in the final answer or at various stages of processing, to discover the source and make corrections.

Decimal Numbering System -- The widely used numbering system having Arabic numerals 0 through 9. It uses 10 marks (0 through 9) thus having a base or radix of 10. In the decimal system we think in "tens." For example, here's how we arrive at the decimal number 2,345:

2000 - 300 - 40 - 5 - 2,345

In the decimal system, all numbers are obtained by using the radix (total number of marks), in this case, 10, raised to various powers.

Decision -- (1) In management, a conclusion arrived at after consideration. (2) In programming, a choice between alternatives depending on prior conditions and use of specified parts of the program.

Delay -- The length of time after either the occurrence of an event for individual event reporting or the close of a reporting period for summary reporting before reports are made available. Delay covers the time needed to process data and to prepare and distribute reports.

Dump -- To record the contents of internal storage at a given instant of time, usually to help detect program mistakes or errors, or to remove a program and data from the processor to permit running another program.

Electronic Data-Processing System -- A machine system capable of receiving, storing, operating on, and recording data without the intermediate use of tabulating cards. The system is also able to store internally at least some instructions for data-processing operations, and to locate and control access to data stored internally.

End-of-file -- Special symbols and a trailer label indicate end of a file. Several short files can be recorded on one tape, whereas a long file may extend over several reels--a multi-reel file. Automatic procedures are used to finish processing when the end of a record file on tape is reached.

End-of-Tape -- A reflective spot or other indicator is placed near the physical end of the tape to signal the end. Automatic procedures are used to handle tapes when the physical end of an input or output tape is reached.

Erase -- To erase information stored on a magnetic tape, magentic drum, or other storage device.

External Memory -- A storage unit such as a magnetic tape which is external to the computer.

File -- One or more records concerning people, things, or places that are closely related and handled together for processing.

File Label -- A record placed before (and after) the records on tape to indicate their nature, when written, when to use, and how long to retain the file.

Master File -- A file of records containing a cumulative history or the results of accumulation; updated in each file-processing cycle, and carried forward to the next cycle.

Transaction File -- The transactions occurring over a period of time and accumulated as a batch ready for processing against the master files that are affected.

Flow Diagram (or Chart) -- A chart showing all the logical steps of a program. A program is coded by writing down the successive instructions that will cause the computer to perform the logical operations represented by a flow chart and necessary for the problem to be solved.

Housekeeping -- The portion of a program which involves the setting up of constants and variables to be used in the program. The housekeeping must be done before any productive work is done by the computer.

Instruction -- A word or part of a word which tells the computer to perform some operation.

Instruction Code -- An artificial language for expressing or describing the instructions which can be carried out by the computer.

Input -- Transfer of external information into the computer.

Internal Arithmetic -- Any computations performed by the arithmetic unit of a computer, as distinguished from those performed by the peripheral equipment.

Internally Stored Program -- A sequence of instructions (program) stored inside the computer in the same storage facilities as the computer data, as opposed to being stored externally on punched paper tape, pin boards, etc.

Interrupt -- The ability of one device to stop the operation of another to indicate readiness to supply or receive data; for example tape units can interrupt the processor when ready to read or write.

Jump -- To transfer control by executing an instruction that specifies the location of the next instruction to be executed by the program. Also called "branch" or "transfer." An unconditional jump is made to occur whenever the jump instruction is encountered in the program. A conditional jump transfers control only if some specified logical condition is satisfied; if the condition is not satisfied, the next instruction is taken in normal sequence.

Key Punch -- (noun) -- A typewriter-like machine for recording data on punched cards by punching a code into them, and, often, printing the same data on the cards.

Label -- In symbolic programming, a name consisting of several alpha-numerics (perhaps required to start with an alpha) to serve as an address while writing a source program; absolute addresses are assigned during assembly or compilation.

Logical Operations -- Nonarithmetical operations such as selecting, search, sorting, matching, comparing, etc.

Magnetic Drum -- Rotating cylinder surfaced with a material that can be magnetized. Used to store information in machine language.

Magnetic-Drum Storage -- A device that stores data on tracks around a rotating cylindrical drum surfaced with a magnetic coating. A magnetic read-write head is usually associated with each track so that the desired track can be selected by electric switching. Data from a given track are read or written sequentially as the drum continually rotates.

Magnetic-Tape Storage --- A storage device consisting of plastic tape or metal coated with magnetic material. A read-write head is associated with each row of bits on tape so that a frame can be read or written at one time as the tape moves past the head.

Management Information System -- A data-processing system designed to supply management and supervisory personnel with information consisting of data that are accurate, timely, and new.

<u>Matrix</u> -- (1) A rectangular array of numbers, subject to mathematical operations, such as addition, multiplication, and inversion, according to specified rules. Any table is a matrix. (2) An array of circuit elements, such as diodes, wires, magnetic cores, and relays, arranged and designed to perform a specified function -- for example, conversion from one number system to another.

<u>Media</u> -- Magnetic tape, punched cards, and punched tape used to hold data and used primarily for input and output.

<u>Merge</u> -- To produce a single sequence of items, ordered according to a certain rule, from two or more sequences previously ordered according to the same rule, without changing the items in size, structure, or total number. Merging is a special kind of collating.

<u>Microsecond</u> -- A millionth of a second.

<u>Millisecond</u> -- A thousandth of a second; one thousand microseconds.

<u>Mnemonic Code</u> -- Instructions for a computer written in a form which is easy for the programmer to remember, but which must later be converted into machine language.

<u>Nanosecond</u> -- A billionth of a second; a thousandth of a microsecond.

<u>Off-Line Equipment</u> -- Equipment not connected directly to the central processor but working through an intermediary device. For example, a processor can write output on an on-line magnetic tape that is later used as input to an off-line printer for printing reports.

<u>Operand</u> -- Any one of the quantities entering into or arising from an operation. An operand may be an indication of the location of the next instruction or a result from computation.

<u>Operation</u> -- A specific action which the computer will automatically perform whenever the instruction calls for it.

<u>Operating System</u> -- That part of a software package designed to simplify housekeeping programming. May include an input-output control system, sort-merge generators, data-conversion routines, and test routines.

<u>Output</u> -- Process of transferring data from internal storage of a processor to some other storage device. A specific output area may be used for organizing data prior to the output operation.

<u>Overflow</u> -- In an arithmetical operation, the generation of a quantity that is too large for the capacity of the register or location that is to receive the result.

Parity Bit -- A bit associated with other data bits to get some specified relation, such as an odd or even total number of bits for each character. A parity bit is usually associated with the frame for each six-bit character on tape; also, parity bits may be placed at frequent intervals to associate them with the seven rows of bits (six for data and one for parity) along the tape.

Parity Check -- A summation check in which the binary digits in a character or word in storage or a character, word, or row on tape, are added (modulo 2) and the sum checked against a single, previously specified digit; for example, a check which tests whether the number of ones is odd or even.

Peripheral Equipment -- Units which work in conjunction with the computer but are not part of the computer itself; e.g., tape reader, analog-to-digital converter, typewriter, etc.

Printer, High-Speed -- High speed printing that makes use of rotating print wheels or a chain with raised type faces and fast-acting hammers to press the paper against the desired character at the instant it is in the correct position.

Process-Time -- The time for translating a source program into an object program through the action of a processor program and a computer.

Processor -- (1) Any device capable of accepting data, applying prescribed processes to them, and supplying the results of these processes. Usually internally-stored program, but may be externally-stored or built-in. (2) An internally-stored-program electronic computer and peripheral equipment used for business data processing. (3) A program used in compiling a source program to produce an object program ready to execute with data.

Program (noun) -- A plan for the automatic solution of a problem. A complete program includes plans for the transcription of data, coding for the processor, and plans for the absorption of the results into the system. The list of coded instructions is called a "routine."

Programmer -- A person who prepares the planned sequence of events the computer must follow to solve a problem, but who need not necessarily convert them into detailed instructions (coding).

Programming -- The process of creating a program; includes applications analysis, design of a solution, coding for testing to produce an operating program, and development of other procedures to make the system function.

Punched Card -- A card of standard size and shape in which data are stored in the form of punched holes. The hole locations are arranged in 80 or 90 columns with a given pattern of holes in a column representing one alphanumeric character. The data content is read by mechanical, electrical, or photoelectrical sensing of the hole positions.

Punched Tape -- Tape, usually paper, in which data are stored in the form of punched holes arrayed in a frame across the tape.

Random Access -- Access to storage under conditions in which each set of data records is directly addressable. Access to data at random in any desired sequence. More commonly used to mean bulk storage with access within several milliseconds to several microseconds for data at any location.

Read -- (1) To copy, usually from one form of storage to another, particularly from external or secondary storage to internal storage. (2) To sense the meaning of arrangements of hardware or visually readable patterns.

Read-Write Head -- A small electromagnet used for reading, recording, or erasing polarized spots on a magnetic surface.

Real-Time Operation -- Processing data in synchronism with a physical process rapidly enough so that results of data processing are useful to the physical operation. Sometimes called "on-line, real-time control."

Run -- The act of processing, under the control of one or more programs, a batch of transactions -- for example the inventory receipts, issues, etc., for the week -- against all the files that are affected to produce desired outputs consisting of updated files and reports.

Sequence -- In sorting, the ordering of items on an element -- for example, records on a selected key -- according to some rules that utilize the processor's collation table.

Serial Number Control -- The control of messages by assigning a number from a master list when it first originates and, perhaps, adding a suffix number from a local list for each point the message passes through to its destination.

Simulation -- An experimental analysis of an operating system by means of mathematical or physical models that operate in a time-sequential manner similar to the system itself.

Software Package -- The programming aids supplied by the manufacturer to facilitate the user's efficient operation of equipment. Includes assemblers, compilers, generators, subroutine libraries, operating systems, and industry application programs.

Sorting -- The arranging of records so that they are in ascending or descending sequence for some data element used as a key.

Storage -- A device capable of receiving data, retaining them for an indefinite period of time, and supplying them upon command.

System -- Any regular or special method or plan of procedure. In a broader context, a system consists of an organization, people, hardware, and procedures that operate together to perform a set of tasks.

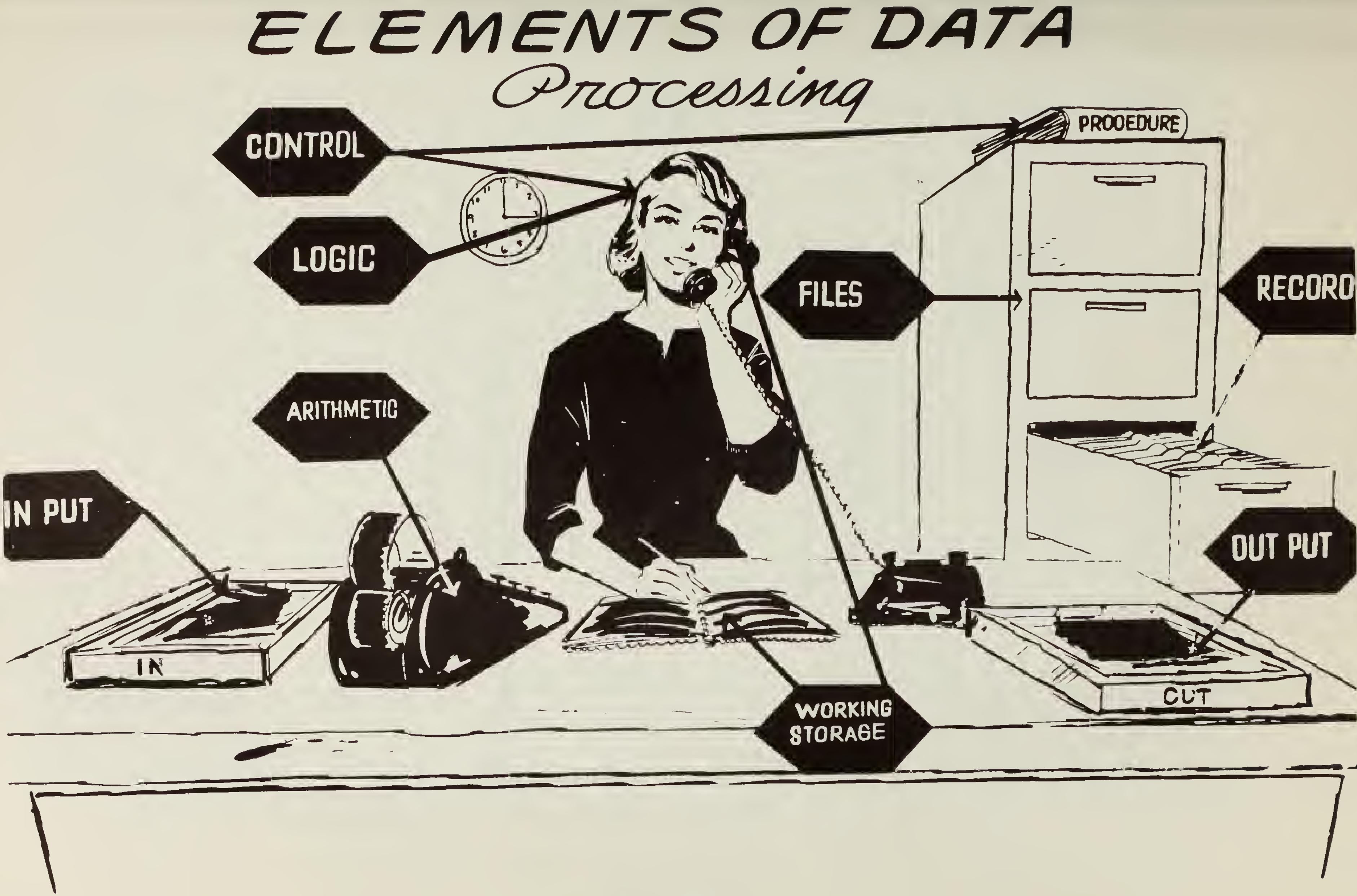
Tape Unit -- A device for reading data from magnetic tape and writing new data (after erasing prior data) on tape. Some tape units read in either direction, although they write in only the forward direction. The device also rewinds tape ready for removal and replacement by another reel.

Transaction -- Event that affects the status of a business -- for example, purchase, sale, issue, and collection. Also called "event."

Transceiver -- Card-reading, modulating, and punching equipment for card-to-card transmission of data over telephone or telegraph grade circuits.

Transition, Rate of -- The period of time during which system changes are made or new equipment introduced. The rate is effected by the degree of change, size of the organization, length of time since the last change, and whether the organization has reached reasonably steady state following the preceding systems change.

Word -- A set of characters occupying one storage location; it is treated by the processor circuits as a unit and transported as such. Ordinarily, the control unit treats a word as an instruction whereas the arithmetic unit treats a word as a quantity.



TRANSPARENCY 1

B -- PROCESSORS

Lesson Plan

COURSE: Automatic Data Processing

SESSION TITLE: Processors

OBJECTIVE:
1. Explain number system
2. Explain data organization
3. Explain operation processor

TIME: 80 minutes

TRAINING AIDS REQUIRED:
Overhead Projector and seven (7) transparencies

Pointer

Movie Projector

Film -- #8969B (Available from Film Library, MCAS, Cherry Point, N. C.

Schematic of data processing components

Processors

In the first session the objective of this automatic data processing was outlined for you; our three-fold objective being:

(1) To build a base

(2) To remove the mystery

(3) To orient you to (substitute name of organization's computer).

By film and demonstration you have been told in broad terms the functions performed by an automatic data processing system which is to accept data, perform prescribed process and store or present data in usable form.

You were briefly introduced to some of the terms which will be a part of our data processing lexicon, and input devices, cards, magnetic tape and punched tape were mentioned.

Also, the relationship of people in the automatic data processing systems was touched on. All of this leads us to the first part of this session.

(TRANSPARENCY #1 ON)

Today we discuss the decimal system people use -- and why. We will discuss the Binary System which computers use and the reasons for this.

The four levels in which data are organized will be shown by slide and discussed.

The last part of today's session, after the break, will be devoted to the basic processor which is the brain or command level of the computer.

Before we discuss today's subject matter, I would like to explain briefly the punched card you received at the first session. Another has been placed at your seat in case you don't have it with you.

Punched-cards are important because data stored on them can be processed electro-mechanically. They are widely used for business data collection. Time clock cards and inventory control cards are two examples in use.

Please take the card with the digits "0" through "9," our alphabet and a few special symbols printed across the top and let's examine it together.

This card is 80 columns wide, has ten punching positions labeled "0" through "9" in each vertical column. Only one hole is punched for each numeral to be represented. Letters use two punches: one in a zone and one in the appropriate column.

You will observe the letters "A" through "I" use one zone: "J" through "R" another: and "S" through "Z" another. Symbols are represented in no specific manner: in fact some are represented with three punches -- for example the comma.

An important fact to remember is that each character, whether it be a digit, a letter or a symbol, has its own individual representation in the punched-card coding scheme.

Punched-cards and how they are used will be covered several times in this course. I explained it to show a similarity to the Binary Coding Scheme which we will discuss in a few minutes.

Any questions on punched-cards?

(TRANSPARENCY OFF)

Of the several number systems, let's discuss the decimal system first -- the one people use in business, in their trade -- and in everyday use.

We like this system because it seems natural. We learned to count on our fingers and some still do.

The origin of this system is reflected by the language of a certain tribe in South America.

Their word for six (6) is "one on the other hand."

Their word for eleven (11) is "one on the foot." As you would surmise, their word for sixteen (16) is "one on the other foot."

What do you suppose their word for twenty (20) is? Yes, it's the word meaning man.

In primitive days this was as far as they could count.

In the decimal system, when counting goes beyond one position, the digit in the next position to the left is increased by one and counting is resumed at "0."

Counting beyond 2 seems easy because of constant practice -- it seems natural that 10 should follow 9, or that 1,000 should follow 999.

In brief, all large numbers make use of the same digits, "0" to "9."

Whenever a digit is written, it has an individual value that is independent of the digits to its right or left -- but -- the overall value of a digit is its individual value multiplied by its position value.

What is position value? It is the number base raised to the power of the digit position occupied.

Let's illustrate this on the blackboard:

(Done at blackboard:)

List the 10 digits in decimal system 0 1 2 3 4 5 6 7 8 9 equals base 10

	Left of decimal point			Right of decimal point	
Position value	10^2	10^1	10^0	10^{-1}	10^{-2}
Y - any digits one may want to use for illustration purposes.	Y x 100	Y x 10	Y x 1	Y x 1/10	Y x 1/100

Overall value of a digit equals its individual value multiplied by its position value.

Our total number is the sum of these products:

(Y x100) + (Y x 10) + (Y x 1) etc.

Gentlemen, I'm sure you must think this elementary, but I needed to illustrate the decimal system to show you a comparison to the Binary System we will see in this film and I will illustrate it also.

The film we are about to see begins with logic -- What is logic? It's sound reasoning. There is logic in the decimal system. Our electronic marvel, the computer, contains some logic. It can compare letters, numerals, or both. But a computer can only think in terms of two conditions: a punch or no punch, pulse or no pulse, a magnetized spot or an unmagnetized spot -- off or on. Therefore, the Binary numbering is ideally suited for computer use as it uses only two digits, "0" and "1."

Let's watch the film now and please pay close attention to Binary coded decimal and alphanumerical representation in which we will demonstrate in detail on the blackboard.

(SHOW FILM)

Automatic processing equipment is designed with many components that can exist in either one of two states. As we said, these two states may be represented in various ways: by a punch or no punch, pulse or no pulse, etc.

This "Yes -- No," or "0 -- 1" condition is popular because it is easier to design equipment that can be put in one of two stable states rather than ten. A light switch is an example of this -- either on or off and stable. Combinations of the two symbols "0" and "1," called Binary Digits or Bits, are used in a variety of ingenious schemes to represent numerals, letters and other symbols.

(ILLUSTRATE AT BLACKBOARD)

		ZONE CHANNEL			NUMERICAL B C D	0 AND 1 PURE		BASE 2	
TAKES	P								
SEVEN	for	Z^2	Z^1		2^3	2^2	2^1	2^0	
BITS	Parity	32	↔ 16	↔	8 ↔	4 ↔	2 ↔	1	= 8
TO	may be								= 16
REPRE-	Odd			0 =		0	0	0	= 64
SENT	or			1 =		0	0	1	
ONE	Even			2 =		0	1	0	
CHAR-				3 =		0	1	1	
ACTER				4 =		1	0	0	
				5 =		1	0	1	
				6 =		1	1	0	
				7 =		1	1	1	
					B C D				
				8 =	1	0	0	0	
				9 =	1	0	0	1	

Organization of Data

Data processing equipment senses, stores and manipulates a wide variety of characters. Individual characters must be grouped together in a way that is practical for use by people and machines.

Data are organized into 4 levels for processing:

(SHOW TRANSPARENCY #2)

1. CHARACTER -- The 64 numerals, letters and symbols we've talked about.
2. ITEM -- Characters grouped to specify a particular unit of information -- a quantity -- an employee number.
3. RECORD -- One or more data items related in some meaningful way -- stock record -- payroll record, etc.
4. FILE -- A group of related records.

There are several schemes for organizing data items for processing, based on the fact that the length of the items may vary greatly.

(SHOW TRANSPARENCY #3)

The variable-length scheme allows any item to be any length without restraint. The end of each item is identified by an item separator used between every two items. Under this scheme, an item is identified not by its location, but by its sequential relationship to other items. To keep the item count correct, this plan requires identification of any items omitted.

Example -- Check Stub

Under the selectable-length scheme any length may be chosen for an item -- say the employee number -- but once chosen all numbers must be

given the same length at the expense of filling out short numbers with zeros. (EXPLAIN CHECK STUB)

The fixed word and block scheme specifies the number of characters that are treated as a word, and the number of words handled as a block. The word and block length are designed into the equipment by the manufacturer, and the user must adhere to his specifications.

We have gone through the decimal system people use because it seems natural and the Binary system computers use because of its two (2) stable states.

We have talked about binary coded decimals, a code using four binary digits for simplicity and the alphanumerical system, which covers all our numbers, letters and symbols, represented by seven (7) bits, including parity.

We have discussed the four (4) levels of data and how they are organized into records and files.

Let's take a break and then we will discuss the actual brain of the computer -- The Processor.

Processors

(SHOW TRANSPARENCY #4)

I am going to describe the basic functions of the processor and its components. I will describe the components briefly and tell how each component performs. I will demonstrate with overhead projections and discussion and then we have a movie which covers the processor operation.

(SHOW TRANSPARENCY #5)

What is a processor? First, the definition of a processor is:

1. Any device capable of accepting data,
2. Applying prescribed procedures,

3. Storing data,

4. Performing arithmetic and logic operations,

5. Controlling the entire operation,

6. Presenting data in usable form.

That was quite a mouthful -- let's go over them again. (REPEAT ABOVE 1 - 6 SLOWLY AND TAKE EACH STEP SEPARATELY.)

What is a processor?

1. Any device capable of accepting data.

a. A person has this capability.

b. An adding machine or calculator also has the ability to accept data when numbers are used.

c. A processor accepts data through input units on punched cards or magnetic tape.

2. Applying prescribed procedures.

a. A person again has this capability.

b. An adding machine or calculator can apply procedures too, when you push the "add" button. It adds --

c. A processor applies procedures by programs fed to the high speed storage. (PROGRAMS WILL BE COVERED LATER.)

3. Storing data.

a. A person can store data in memory.

b. An adding machine again qualifies by storing each number inserted into it.

c. A processor has the capability to store large volumes of data also. How it does this will be explained in detail in your next session.

4. Performing arithmetic and logic operations.

a. A person can perform simple arithmetic and logic in his brain from memory.

b. Now an adding machine doesn't qualify in this category -- it cannot do anything but add so cannot perform logic.

c. A processor performs arithmetic functions and logical operations by using electrical pulses with binary code -- in millionths of a second.

5. Controlling the entire operation.

a. A person controls entire operations by brain power and memory and reference by communication of sight and sound which could be instructions, procedures for sight and telephone for sounds.

b. The processor controls the entire operation by programs previously inserted in the processor -- this is controlled by sequence operations that are performed in millionths of a second. This will be explained in the Programming Session.

6. Presenting data in usable form.

a. Data in usable form is the final output of any processor.

(1) A person can present data in usable form prepared slowly and manually.

(2) The processor presents data in usable form by punched cards, magnetic tape or in printed form.

Incidentally, these usable forms will be presented in your next two sessions.

A <u>basic processor</u>, consists of a <u>control unit, arithmetic and logic unit</u>, and a <u>high speed storage unit</u>.

If you will note on the schematic before you, each component I mentioned -- is in the Basic Processor. We will confine our comments to the part enclosed in red.

Let's start with the High Speed Storage Unit.

(SHOW TRANSPARENCY #6)

1. The high speed storage unit is used to store programs and data.

Programs and data are fed to high speed storage through input units and stored in separate locations.

2. It is used to receive data from input units.

High speed storage is used to receive data from input units for storage -- through control of the control unit. The control unit notifies high speed storage when to accept data and where to store.

3. It supplies data and instructions to the Control Unit and to the Arithmetic Unit during processing.

Data could be permanent storage or temporary storage that was recently received from input units. Instructions are normally in permanent storage, and are taken in sequence from a program.

4. High Speed Storage furnishes results to the output units.

Results furnished to output units are directed by the control unit, to a specific output and in a specified sequence.

5. High speed storage varies in capacity from a few thousand to many thousand characters -- and each character is available in microseconds (millionths of seconds).

If you'll look at your handout, you will see Bulk Storage. It is also available to the processor for instruction and data but at a slower speed.

Now, let's go to the Arithmetic and Logic Unit.

(a) Arithmetic Functions performed are --

addition, subtraction, multiplication, and division

(b) Logical Functions are performed such as comparing data --

(1) Compare for Numeric Sequence.

If you will recall the last hour that "binary" coding was discussed -- numerics consisted of four "bits."

Let's take the digit - 2 - and put it in binary code.

0 0 1 0

Now let's take the digit -3- and put it in binary code.

0 0 1 1

By comparing the binary value of the two digits -- the processor can determine which character is earlier.

(2) Data can also be compared in alphabetic sequence --

Let's take the two four bits we have for numerics and make alphabetic characters out of them -- by adding 0 and 1 to the zones -- we have a "B" character.

0 1 0 0 1 0

Now, using the same bits in the next group of bits we have the "C" character.

0 1 0 0 1 1

The binary value of these bits can also be compared to see which is earlier or later. The same procedure applies to comparing numeric and alphabetic combinations in sequence.

These data items consisting of letters, numerals, or both, can be compared, in order to find which is earlier or later in the alpha-numeric sequence by comparing the binary value of each character.

HOW DOES IT DO THIS?

Through logical circuitry which we will not go into at this time.

(c) Now we come to the most important unit of the processor -- the Control Unit.

(1) This unit controls the flow of data through the system and initiates desired arithmetic or logical operations.

(2) It controls the circuitry that selects each instruction in proper sequence.

(3) It interprets each instruction -- ADD, STORE, MOVE, etc.

(4) Finally it causes all other parts of the processor to carry out the actions specified in an instruction.

(d) Let's compare a processor to the picture that was shown in the first session -- of a girl sitting at a desk.

(SHOW TRANSPARENCY #7)

First let's define a processor again -- a processor is any device capable of the following:

ACCEPTS DATA	Girl's Brain -- Input Basket	Control Unit -- High Speed Storage
APPLY PRESCRIBED PROCEDURES	Written Procedures and that retained in memory	Available in high speed storage in microseconds
STORAGE	Brain (Memory) File Cabinets, Scratch Pad	Available in high speed storage in microseconds
PERFORM ARITHMETIC & LOGIC OPERATIONS	Adding Machine	Arithmetic & Logic Unit
CONTROL ENTIRE OPERATION	Girl's Brain	Control Unit
PRESENT DATA IN USABLE FORM	Output Basket -- Manual preparation; Handwritten, Typewritten, Telephone	Magnetic Tape: 33,333 char. per sec. Punched Cards: 300 cards per sec. Printed Matter: 1000 lines per min.

HOW ARE THESE FUNCTIONS INITIATED?

By Instruction Routines.

(SHOW TRANSPARENCY #6)

(1) Instruction routines selected by the Control Unit causes other components to carry out the necessary operations.

(2) A list of coded instructions, called an instruction routine or program routine -- tells the Control Unit what to do and what sequence to follow.

(a) An instruction might say -- "ADD X to Y," "READ" or "STOP."

1. Each instruction consists of an operation code -- ADD, SUBTRACT, etc., and the address of data in storage.

(b) Each instruction tells the Control Unit what operation to perform -- which data to use -- and where to place results.

Instructions or programs are stored in High Speed Storage until called for by the Control Unit.

THEN WHAT HAPPENS?

The instructions and data are in High Speed Storage -- they go to areas called <u>registers</u>.

There are many types of Registers such as,

1. Instruction Register

2. Address Registers

SAMPLE INSTRUCTION -- "ADD 2000" (WRITE ON BLACKBOARD -- ADD 2000.)

Might mean "add the contents of High Speed Storage location 2000 to the contents of a special area where sums are formed."

Now let's follow the overall control and operation steps for the processor.

Look at your schematic.

Under guidance of the control unit, the input units read data for transfer into specified locations of the high speed storage unit -- and is stored until the control unit requests it.

1. The Control Unit obtains an instruction from High Speed Storage and interprets -- "ADD"

(ADD)

2. The Control Unit then sets up Arithmetic and Logic circuitry as specified by the instruction.

3. The Control Unit obtains data from high speed storage -- executes operation and sends to Arithmetic and Logic Unit.

4. The Arithmetic and Logic Unit then performs the specified operation on the data and returns the results to high speed storage via the Control Unit -- for storage until the Control Unit requests the results, and notifies the Control Unit that the operation is complete.

5. The Control Unit interprets the output instruction to send the results from storage to an output unit -- which in turn presents data in usable form, such as: punched cards, magnetic tape, printer.

Registers

1. A Register is a device capable of receiving information, holding it, and transferring it as directed by Control Circuits.

Registers are required by the Control Unit to know at all times what process is in operation and what the next procedure will be.

Stop and think a moment -- each process is taking place in less than one second intervals -- and some in billionths of a second.

2. Registers are named according to their function.

a. An Instruction Register -- contains the instruction being executed.

b. A Data or Address Register -- holds the address of the storage location specified by the instruction.

c. An Instruction Counter -- stores the address of the next instruction to be executed.

d. The Decoder -- translates the operation code part of an

instruction by setting up the appropriate arithmetic and logic circuits for its performance.

e. An Accumulator -- located in the arithmetic unit is a register to form sums and other arithmetic results.

EXAMPLE:

(1) The Control Unit instructs the equipment to perform the following:

(a) Copy the first number, or operand, from a storage location into the Accumulator.

(b) Get the second number from a storage location and add the number in the accumulator.

(c) Continue to add/or copy the accumulator content into a storage location.

(MOVIE)

Now we will look at a film showing the operation of the processor and the component parts and processing functions.

The film describes the 3 parts of the processor -- Storage, Control and Arithmetic and Logical Parts.

It starts with the Storage Unit. In the first frame you see, will be the Storage Unit. It is talking about High Speed Storage within the Processor. Then it goes on to the Arithmetic and Control Units.

SUMMARY

We have seen that the Processor is the brain of the Arithmetic Data Processing System.

The Processor consists of three basic components which are --

1. High Speed Storage Unit -- to store data and instructions.
2. Arithmetic and Logic Unit -- to perform calculations.
3. Control Unit --
    a. To direct the flow of data.
    b. Initiate desired operations.
    c. Control output.

Remember the special Registers are used to inform the Control Unit what instruction is in process and what the next instruction number is.

HANDOUTS

Earnings Statement

213 9033	18400	00000	00000	14559	1196	01484	0248		125	00788	04	24	65
EMPLOYEE NUMBER	REGULAR	OVERTIME	OTHER	NET PAY	CSRA/FICA	FED. W/H TAX	STATE TAX		LIF INS.	FIXED DED.*	MO.	DAY	YEAR
	EARNINGS				DEDUCTIONS						PAY PER. ENDING		

Form approved by
Comptroller General, U.S.
May 9, 1958 (Rev. 10-61)

NAVCOMPT FORM 906 (REV. 10-61)

EMPLOYEE'S STATEMENT OF EARNINGS AND DEDUCTIONS

EMPLOYEE NUMBER	EARNINGS: REGULAR	OVERTIME	OTHER	NET PAY

CSRA / FICA	FED. W/H TAX	STATE TAX		LIFE INS.	FIXED DED. *	MONTH	DAY
DEDUCTIONS						PAY PER. END.	

INFORMATION ONLY

RETAIN THIS STATEMENT AS A PERMANENT RECORD OF YOUR EARNINGS AND DEDUCTIONS

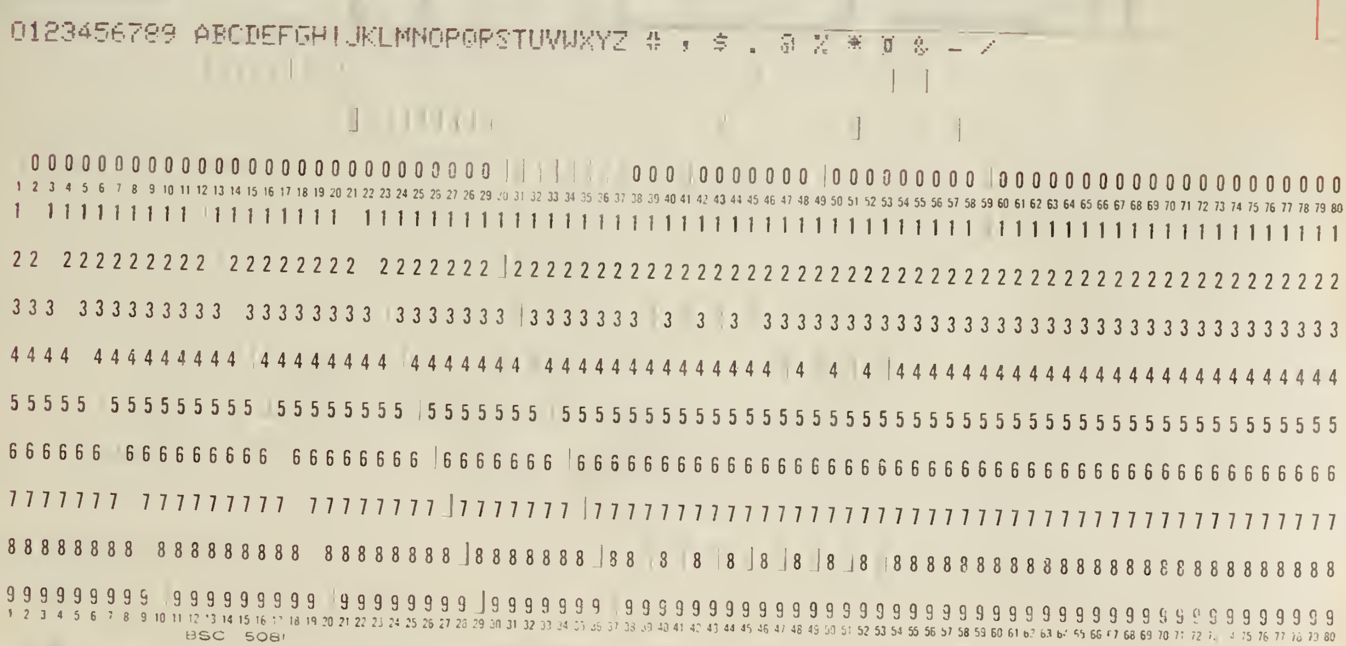
NON-NEGOTIABLE

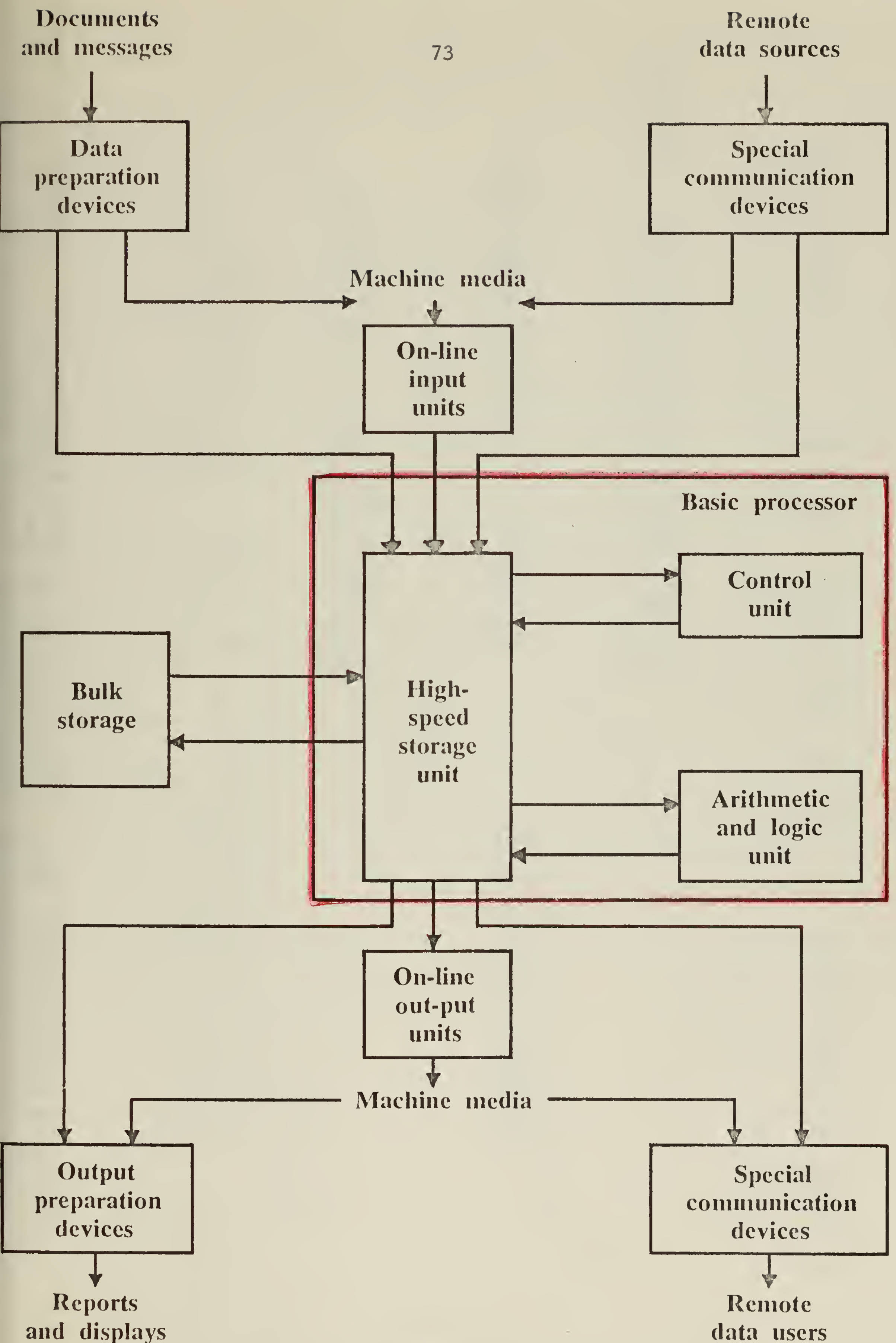
* INCLUDES BONDS, HEALTH INSURANCE AND OTHER DEDUCTIONS

EMPLOYEE NUMBER	REGULAR	OVERTIME	OTHER	NET PAY	CSRA/FICA	FED. W/H TAX	STATE TAX		LIFE INS.	FIXED DED.	MO.	DAY	YR.
	EARNINGS				DEDUCTIONS						PAY PER. END.		

D32712 TABCO

Card Coding Scheme





Schematic of major components of a data-processing system.

NUMBER SYSTEMS

DECIMAL - The System People Use

BINARY - The System Computers Use

ORGANIZATION OF DATA

TRANSPARENCY 1

FIRST HALF SESSION TITLE

ORGANIZATION OF DATA

CHARACTER = 0-9, A-Z, ETC

ITEM = A GROUP OF RELATED CHARACTERS

A QUANTITY.

AN EMPLOYEE NUMBER, ETC.

RECORD = A GROUP OF RELATED ITEMS.

A STOCK RECORD.

A PAYROLL RECORD, ETC.

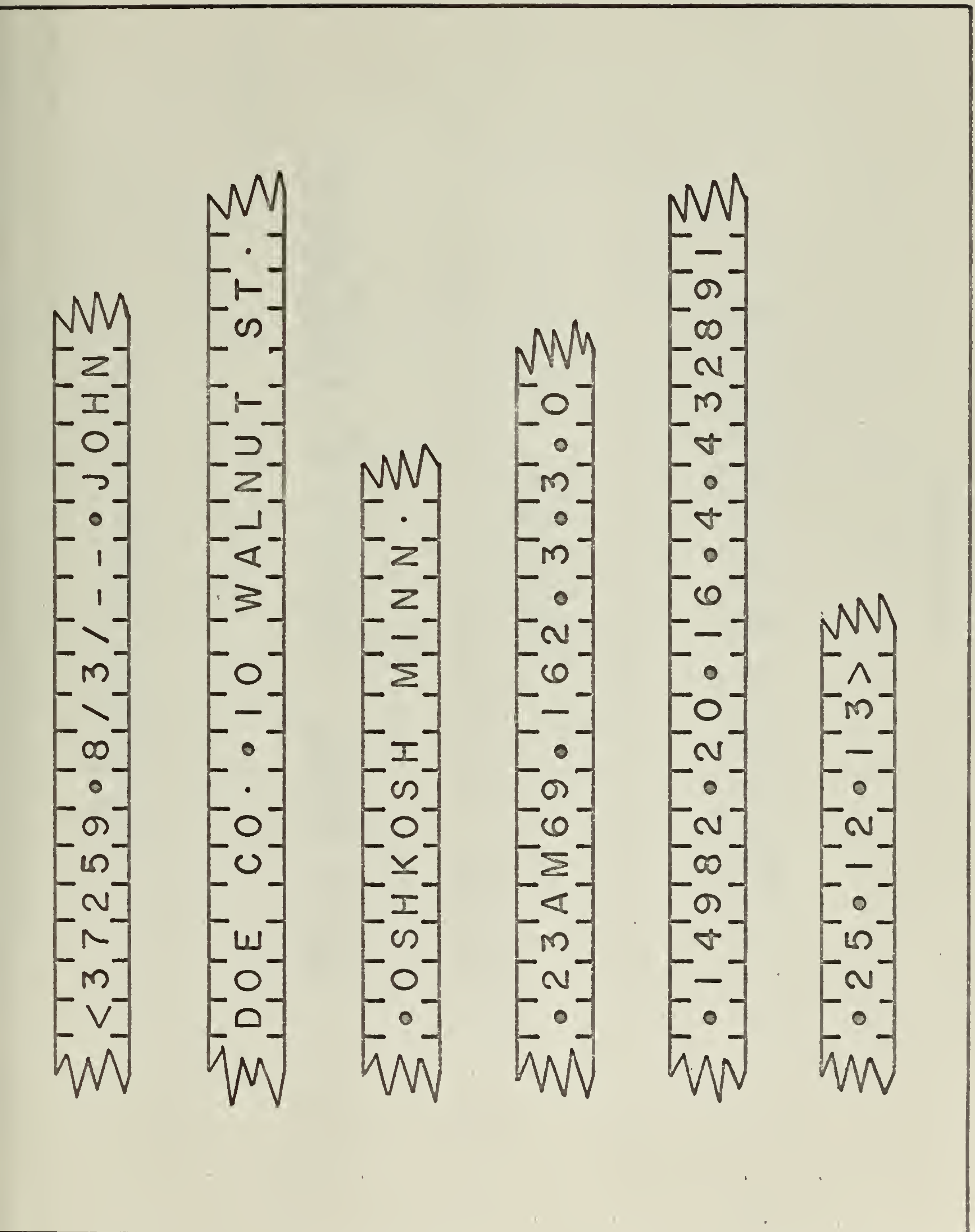
FILE = A GROUP OF RELATED RECORDS.

A SUPPLY FILE

A PAYROLL FILE, ETC.

TRANSPARENCY 2

VARIABLE LENGTH SCHEME



TRANSPARENCY 3

PROCESSOR

BASIC FUNCTIONS

AND

COMPONENTS

TRANSPARENCY 4

SECOND HALF SESSION TITLE

A PROCESSOR . . .

Accept data

Applies prescribed procedures

Stores data

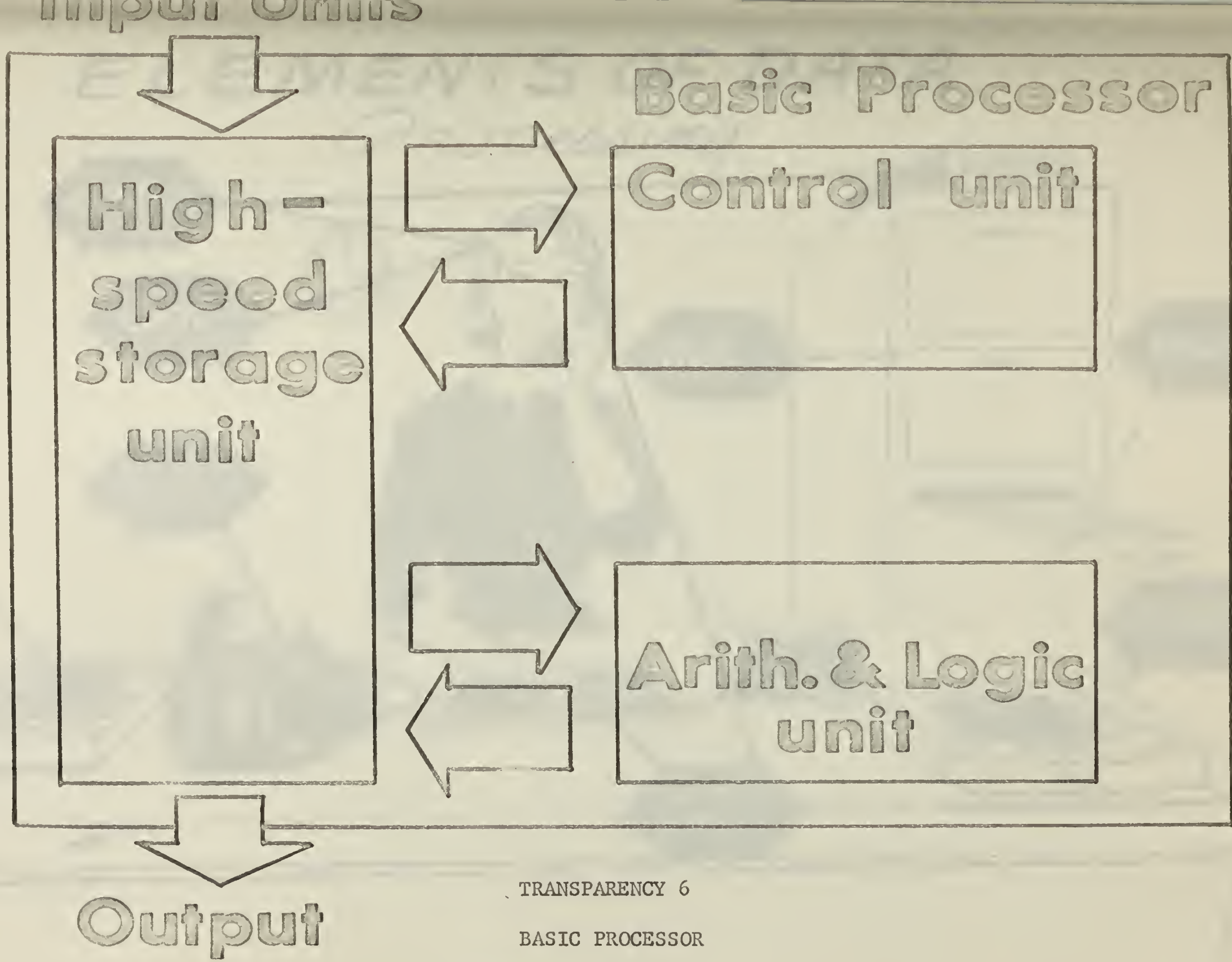
Performs arithmetical & logical operations

Controls operations

Presents data in usable form

TRANSPARENCY 5

DEFINITION OF A PROCESSOR

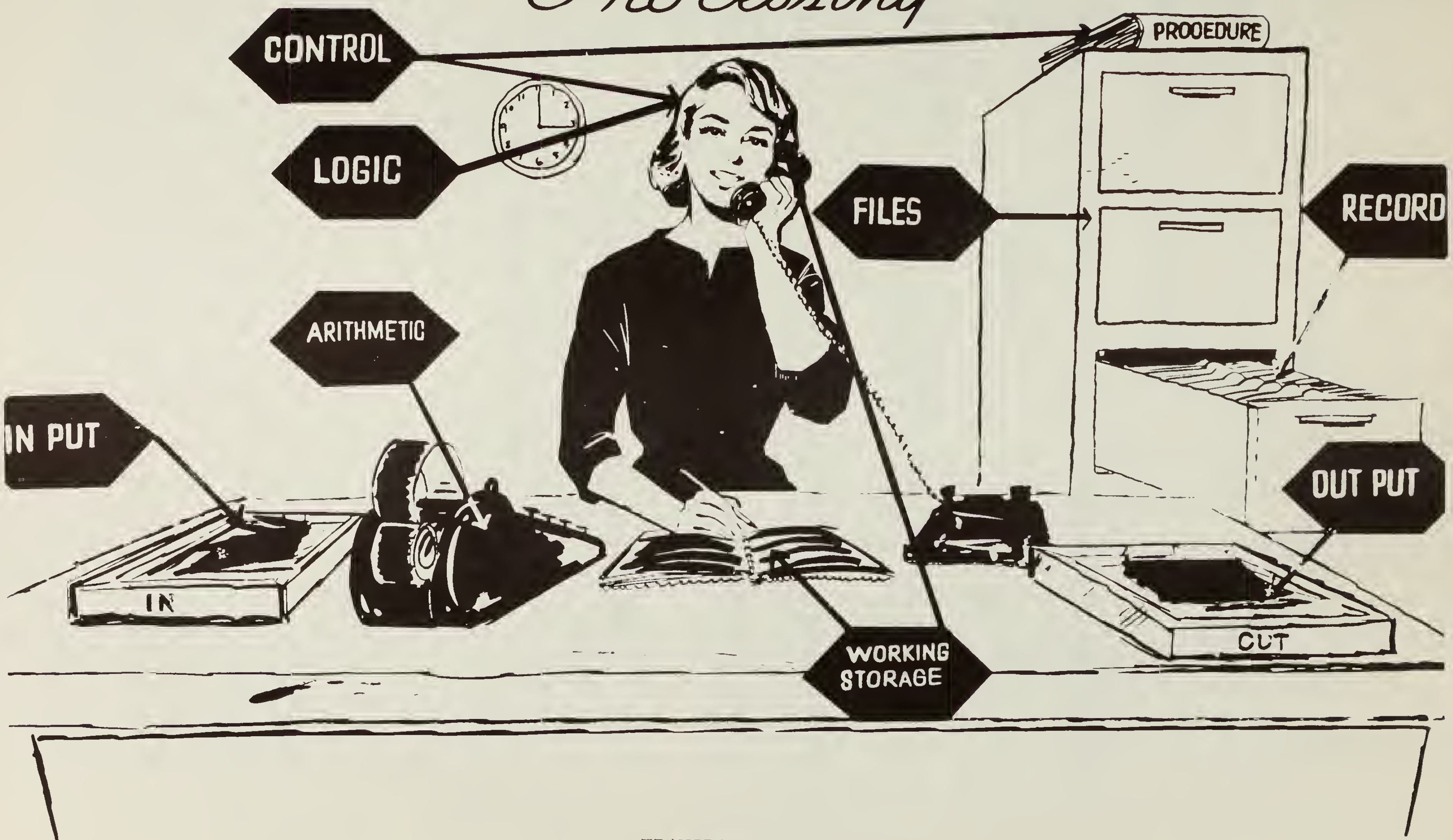


TRANSPARENCY 6

BASIC PROCESSOR


Arithmetic Logic
Unit
Output

ELEMENTS OF DATA *Processing*



TRANSPARENCY 7

C -- STORAGE DEVICES

COURSE: AUTOMATIC DATA PROCESSING

SESSION TITLE: STORAGE DEVICES

OBJECTIVE: To discuss computer storage devices and equipment

TIME: 50 minutes

TRAINING AIDS REQUIRED: Overhead projector and 8 transparencies

Pointer

Film, "Storage Devices," Mn 8969B (Available from Film Library, MCAS, Cherry Point, N. C.)

7 Film slides (Available from Film Library, MCAS, Cherry Point, N. C.)

Facsimile of a magnetic card.

Plastic disc containing magnetic core.

Coil of punched paper tape.

Strip of magnetic tape.

of high speed storage, CORE STORAGE, works. We will also have a look at the various storage devices, and discuss some of their characteristics.

First, let us turn our attention to the types of storage, and discuss their use.

(SHOW TRANSPARENCY #1)

Punched-Cards. Punched-cards are used to store information such as a supply file, a payroll record, etc. These are good for permanent bulk storage in some cases, but are not used for high speed storage.

Punched Paper Tape. Punched paper tape is used to store information in punched form. It is similar to cards. It is generally used when data length exceeds the 80 column limitation of punched-cards.

Magnetic Tapes, Cards and Discs. These are used to store information similar to file cabinets in your office. Characters, items, records and files may be stored on these devices.

Remember that an Item is:

A group of related characters, such as a quantity or an employee number.

A record is a group of related items such as a stock record, payroll record, etc.

A file is a group of related records such as a supply file, a payroll file, etc.

Magnetic Drum. This is a high speed type of storage, used in some computers for internal storage. In most computers, however, magnetic drum is used to store overflow data from the processor and to store data that is required often.

Magnetic drums are used in some processors to store programs, and

hold registers. Some computers use drums to store bulk information similar to tapes and cards and discs. The use of drums varies with each system design.

Magnetic Core. Magnetic core is commonly used in the processor. It stores instructions to the processor, stores programs, registers and other data. We refer to this as high speed memory or high speed storage. Any of these, except core, may be used for bulk storage.

The use of various storage devices is dependent upon the type of equipment that may be available for recording and reading the storage devices.

Now, let's go back to this list of storage means and describe each type.

Punched-Cards -- (either a 51 or 80 column card.) You are all familiar with the card.

Punched Paper Tape -- Punched paper tape may be five, six, seven or eight channel paper tape. Five channel is used on all standard telegraph equipment. I will pass around a sample.

(SHOW TRANSPARENCY #2)

Magnetic Tape is plastic coated on one side with a metallic oxide. It is 1/2 to 1 1/2 inches wide and usually in reels from 1500 to 3600 feet long. It is similar to the tape used for home recording. Information is permanent and can be retained for an indefinite period of time. It can be read and re-read as many times as necessary. Data is stored on tape in binary bits, tiny magnetized spots. It cannot be seen with the eye. Data is written on and read from magnetic tape by a read-write head. (PASS OUT TAPE.) A read-write head is simply an electromagnetic device which can

magnetize a tiny area on the surface of the tape.

(SHOW TRANSPARENCY #3)

Conversely, it can also sense the presence of a magnetic field. This gives it the ability to read or write. The direction of current flow determines the polarity of the magnetic spot. Consequently, spots can represent ones or zeros, the two digits used for binary recording.

(SHOW TRANSPARENCY #4)

Magnetic Cards. One version of a magnetic card is a flexible card 4½ inches wide X 16 inches long. Only one side of the card is coated with metallic oxide and data is recorded on this side in the form of bits, the same as on tape. (PASS OUT CARD) Cards may be likened to short strips of tape. These cards are housed in magazines. One manufacturer makes a magazine which holds 256 cards. Each card is pulled out of the magazine and passed by a read-write head for reading and writing data. These cards may be individually addressed, and have some of the characteristics of tape storage and some of the next type which is drum storage.

(SHOW TRANSPARENCY #5)

Magnetic Drum is a metal cylinder varying in diameter from 4 to 12 inches. It is coated with magnetic material. This coating is the actual storage medium. This is similar to a log with tape wrapped around it. These drums revolve from 3,000 to 17,000 revolutions per minute depending on the manufacturer. Data is recorded by means of magnetic spots, similar to tape and cards. Read-write heads are mounted in the drum housing and the drum rotates past them.

(FLIP OVERLAY)

As the particular character represented by the bits passes the read-write head, information is written or read out. The head traverses to read each band.

(FLIP OVERLAY)

To speed up the process more read-write heads may be added. This reduces the time to find any particular data.

(SHOW TRANSPARENCY #6)

Magnetic Disc Storage. The magnetic disc is a thin metal disc varying from 1 to 48 inches in diameter, coated on both sides with a magnetic recording material. Information can be stored on either or both sides in the form of magnetic spots. These discs are stacked in units similar to a juke box and are mounted on a rotating shaft. Most of these units vary to the number of discs per unit. The typical unit would contain 50 discs. It is read from and written on by read-write heads on arms which go between each of the discs. These arms move in and out and up and down.

(SHOW TRANSPARENCY #7)

Now, the last type of storage we will discuss is Magnetic Core.

Magnetic Core is a tiny doughnut-shaped ring of iron (or ferromagnetic material), a few hundredths of an inch in diameter (or about the size of a pinhead) which is capable of retaining either one of two magnetic states. + - ; on - off, zero one; 01. Each magnetic core holds one bit. These cores are arranged in a manner so that they can store information.

Now to show you how core storage works, we have a film.

As the particular character represented by the bits passes the read-write head, information is written or read out. The head traverses to read each band.

(FLIP OVERLAY)

To speed up the process more read-write heads may be added. This reduces the time to find any particular data.

(SHOW TRANSPARENCY #6)

<u>Magnetic Disc Storage</u>. The magnetic disc is a thin metal disc varying from 24 to 48 inches in diameter, coated on both sides with a magnetic recording material. Information can be stored on either or both sides in the form of magnetic spots. These discs are stacked in units similar to a juke box and are mounted on a rotating shaft. Most of these units vary to the number of discs per unit. The typical unit would contain 50 discs. It is read from and written on by read-write heads on arms which go between each of the discs. These arms move in and out and up and down.

(SHOW TRANSPARENCY #7)

Now, the last type of storage we will discuss is <u>Magnetic Core</u>.

<u>Magnetic Core</u> is a tiny doughnut-shaped ring of iron (or ferro-magnetic material), a few hundredths of an inch in diameter (or about the size of a pinhead) which is capable of retaining either one of two magnetic states. + - ; on - off, zero one; 01. Each magnetic core holds one bit. These cores are arranged in a manner so that they can store information.

Now to show you how <u>core storage</u> works, we have a film.

It starts with magnetic tape storage, and discusses drum, disc and core storage. It illustrates how core storage works, and how it is used in the processor for high speed storage.

(SHOW FILM)

I will pass around a sample of core. These are the little doughnuts which are strung on wires to make a core matrix.

Now you will recall that each one of these doughnuts, or cores, holds one bit of information. Therefore, it may take seven of these cores to represent one character. For example:

(SHOW TRANSPARENCY #8)

> The letter A is made up of seven bits. A parity bit, 2 zone bits, and 4 numerical bits. These are represented by plus or minus pulses in high speed storage. Now applying these voltages to the cores, we see that if we have a stack of seven cores, we can represent the letter A. Any other number, letter or special character may be represented by changing the bit configuration. This is in fact what is done, the matrices are stacked in an array; in this case, 7 high. In this way we are able to represent any of 64 numbers, letters or special characters. In turn, any number of characters may be combined to form items, records and complete file. How many characters can we store in this array? (Answer: 100.)

Core storage is very fast which makes it desirable for processors. It is also very expensive, as each core has to be strung by hand and the whole matrix and array is hand-wired.

Now let's talk about each of the various storage devices and discuss some of the characteristics of each.

(SHOW SLIDE #1)

This is a Magnetic Tape Station. The tape is wound on reels. The reels of tape vary but there is some sort of take-up mechanism to prevent the tape from breaking. The tape is fed across a read-write head.

Data is represented in the Binary System in the form of magnetic spots.

Addressing: Records on magnetic tape are non-addressable. (EXPLAIN) A tape label is used to identify the record or records on a tape. In order to find a particular address on the reel of tape it may be necessary to pass the location several times.

Data is written, transferred or read by bit, serial or parallel, depending on the equipment. You will recall that serial mode of operation means that one bit of information is written, read, or transferred at a time. Parallel operation, on the other hand, means that several bits or characters are transferred simultaneously.

Capacity: A 2400 ft. reel of half-inch tape can store about 3 to 16 million characters, but is generally less than maximum because of inter-block gaps -- of 3/4 to 2 inches. This allows the tape unit to start and stop and not lose data. Inter-block gap occurs because the tape unit has to start and get up speed. When it stops, it has to decelerate. Although this is very rapid, it still takes time and the unit cannot read or write during this time. A 2400 foot reel of tape with 200 characters per inch holds as many data as 25,000 eighty column cards. Therefore an unlimited amount of information can be stored on tape using more than one reel of tape.

Access Time: Access to data stored on tape is dependent on several elements. An average reading time for a 2400 foot tape at maximum speed may take four and one-half minutes. Tape movement speed ranges from 75-200 inches per second.

Erasable: Tape is erasable and can be re-used many, many times. The information does not disappear in case of power failure.

(SHOW SLIDE #2)

Magnetic Cards

Data is represented in the Binary System in the form of magnetic spots.

Addressing: Each card is individually addressable. Notice the notches on the top and bottom edges. By coding these notches, each card can be made unique. Through a system of rods the card can then be selected in much the same manner as the old key sort system. You can think of the telephone list finder on your desk.

Mode: Either Serial or Parallel.

(SHOW SLIDE #3)

Capacity: One card version has a capacity of 166,400 characters. One machine uses cards and can store over 4 billion characters.

(SHOW SLIDE #4)

Access Time: Very rapid. It depends on the speed of the selector that moves the card to the read head. For example, in this type the magazine. (POINT OUT ON SCREEN) The read-write head is -- (POINT OUT ON SCREEN). The card must be removed from the magazine, moved to the read-write head and replaced in the magazine. Obviously, the card located

closest to the read-write head can be read more quickly and returned to the magazine than one from the far end of the device. A general time is 300 to 500 milloseconds. This is dependent upon the location of the card in the equipment because the card has to be moved to the read head.

Erasable: Yes. These magazines are removable and interchangeable and, like tape, do not disappear in event of power failure.

(SHOW SLIDE #5)

Magnetic Drum

Data is represented in the Binary System in the form of magnetic spots.

Addressing: Storage on a drum is addressable by assigning certain locations on the drum to specific data. However, it need not be addressable.

Mode: Serial and/or Parallel.

Capacity: Magnetic drums in use today vary greatly in capacity. Capacities range from 10,000 to 500,000 numeric characters. They range in diameter from 4 to 12 inches and rotate at speeds of 3000 to 17,000 rpm.

Access Time: Access time varies from a few microseconds to 50 milliseconds.

Erasable and non-volatile.

(SHOW SLIDE #6)

Magnetic Disc

Data is represented in the Binary System in the form of magnetic spots.

Addressing: Addressable in much the same manner as drum storage by assigning certain areas on the disc to specific data.

Mode: Serial and Parallel.

Capacity: Depends on manufacture. A single disc may store as many as 700,000 characters on a single side. These discs may be stacked. Say a -- maximum of 50 discs -- total capacity may be 56 million characters in one unit.

Access time: Varies from 10 milliseconds to several seconds dependent upon the equipment.

Erasable and non-volatile.

(SHOW SLIDE #7)

Magnetic Core

This slide shows a processor. The magnetic core (POINT) is here. (POINT) -- The cover has been removed to show the core.

Data is represented in the Binary System in the form of magnetic spots.

Addressing: Each location in core storage has a unique address so that it can be located. Electronic addressable: no mechanical movement.

Mode: Serial or Parallel.

Capacity: From a few hundred characters to several hundred thousand.

Access Time: In the range of a fraction of a microsecond to 20 microseconds.

Erasable and volatile.

Punched-Cards

Data is represented by punched holes; square or round; can be binary coded but usually some sort of code.

Addressing: Non-addressable.

Mode: Serially.

Capacity: 80 characters.

Access time: Various.

Erasable: No. Card cannot be erased.

Punched Paper Tape

Data Representation: Punched in various methods -- punched holes, code and binary.

Addressing: Non-addressable.

Mode: Serially.

Capacity: Only limited by the length of the tape.

Access Time: Varies due to scanning to locate information.

Erasable: No.

Gentlemen, that concludes our presentation on storage devices. We have introduced the various means used for information storage and we have discussed their uses. We have looked at the equipment used, and we have talked about the characteristics of each.

TYPES OF STORAGE

MAGNETIC ..

TAPE

CARDS

DRUM

DISC

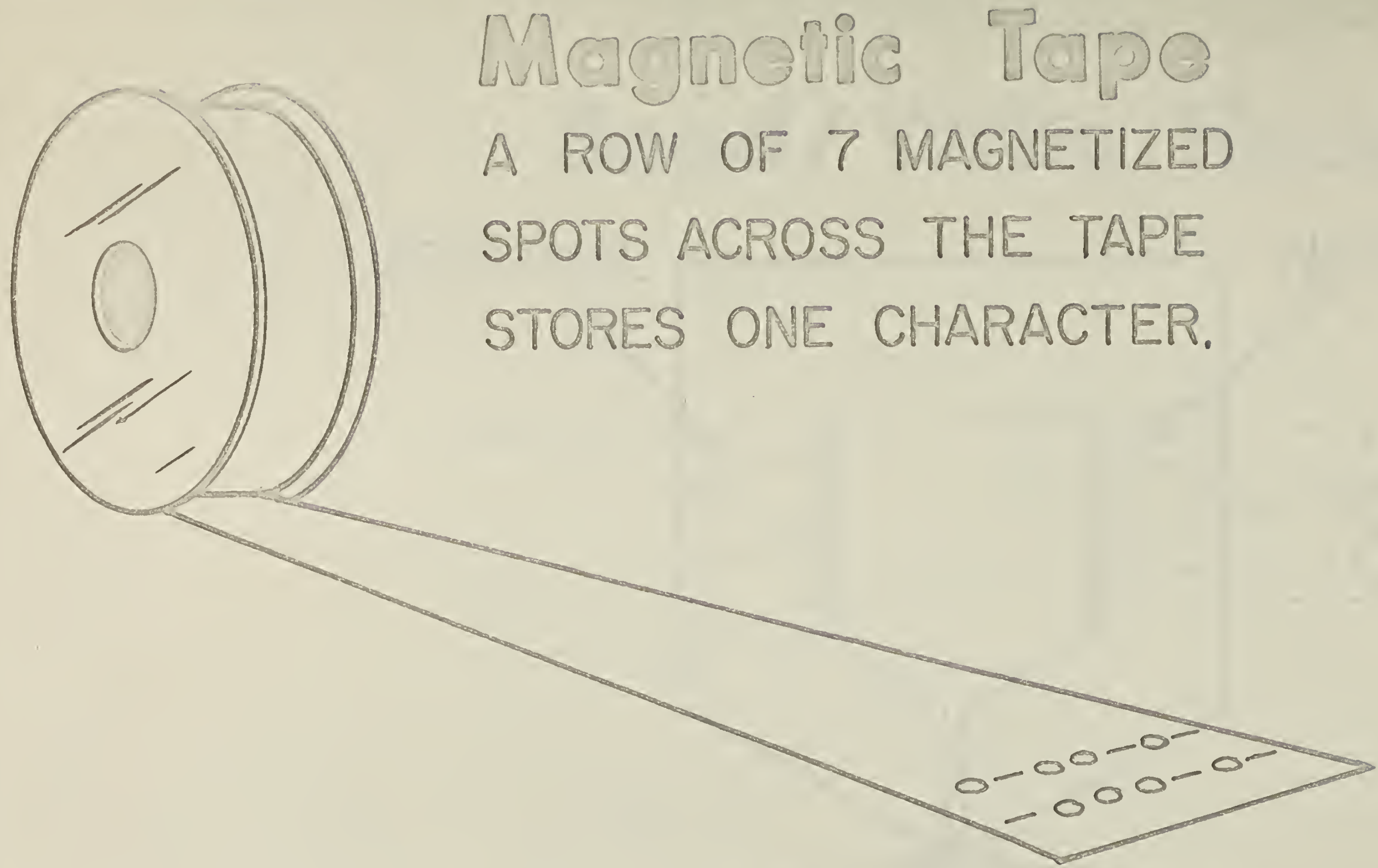
CORE

PUNCHED ..

CARDS

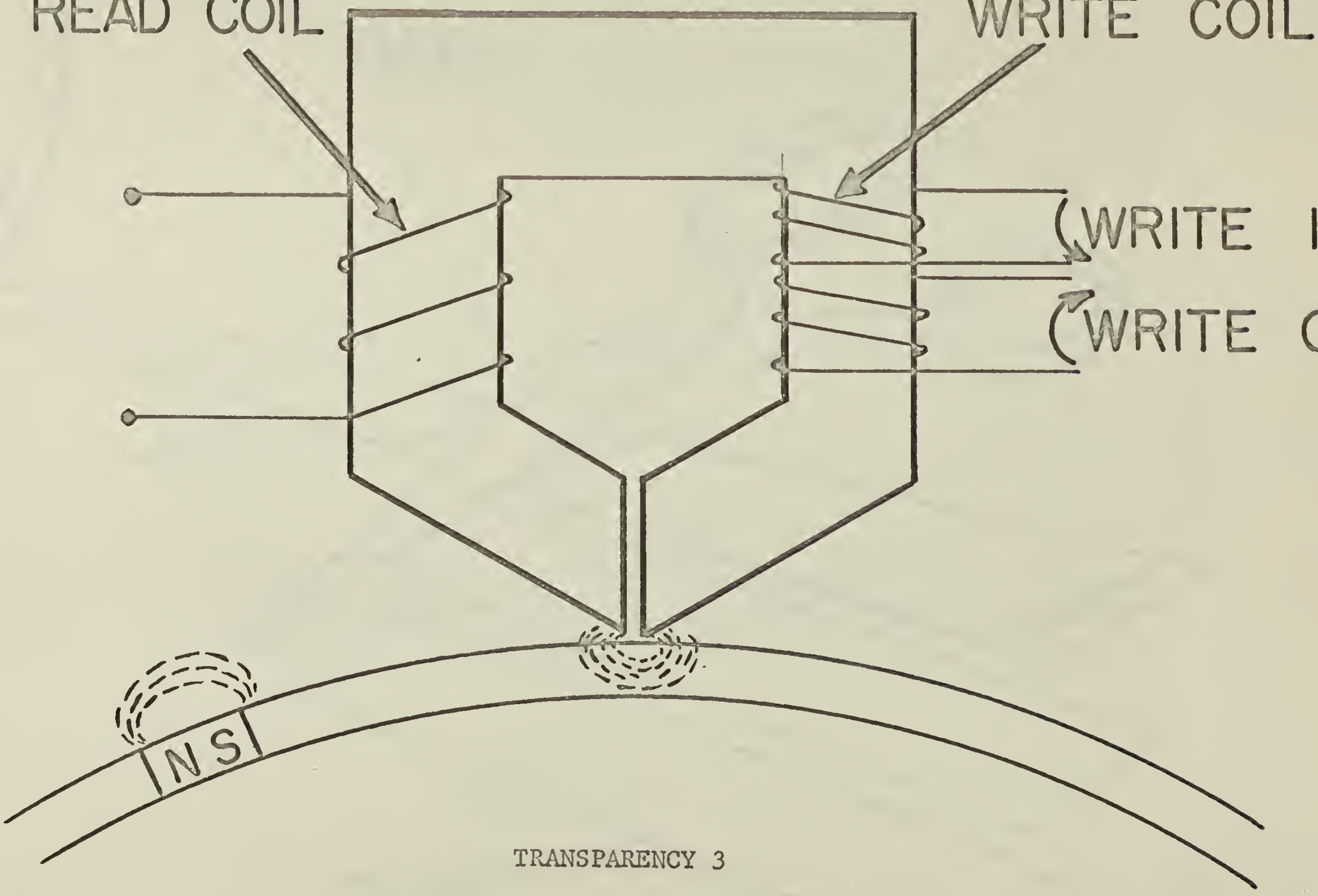
TAPE

TRANSPARENCY 1



TRANSPARENCY 2

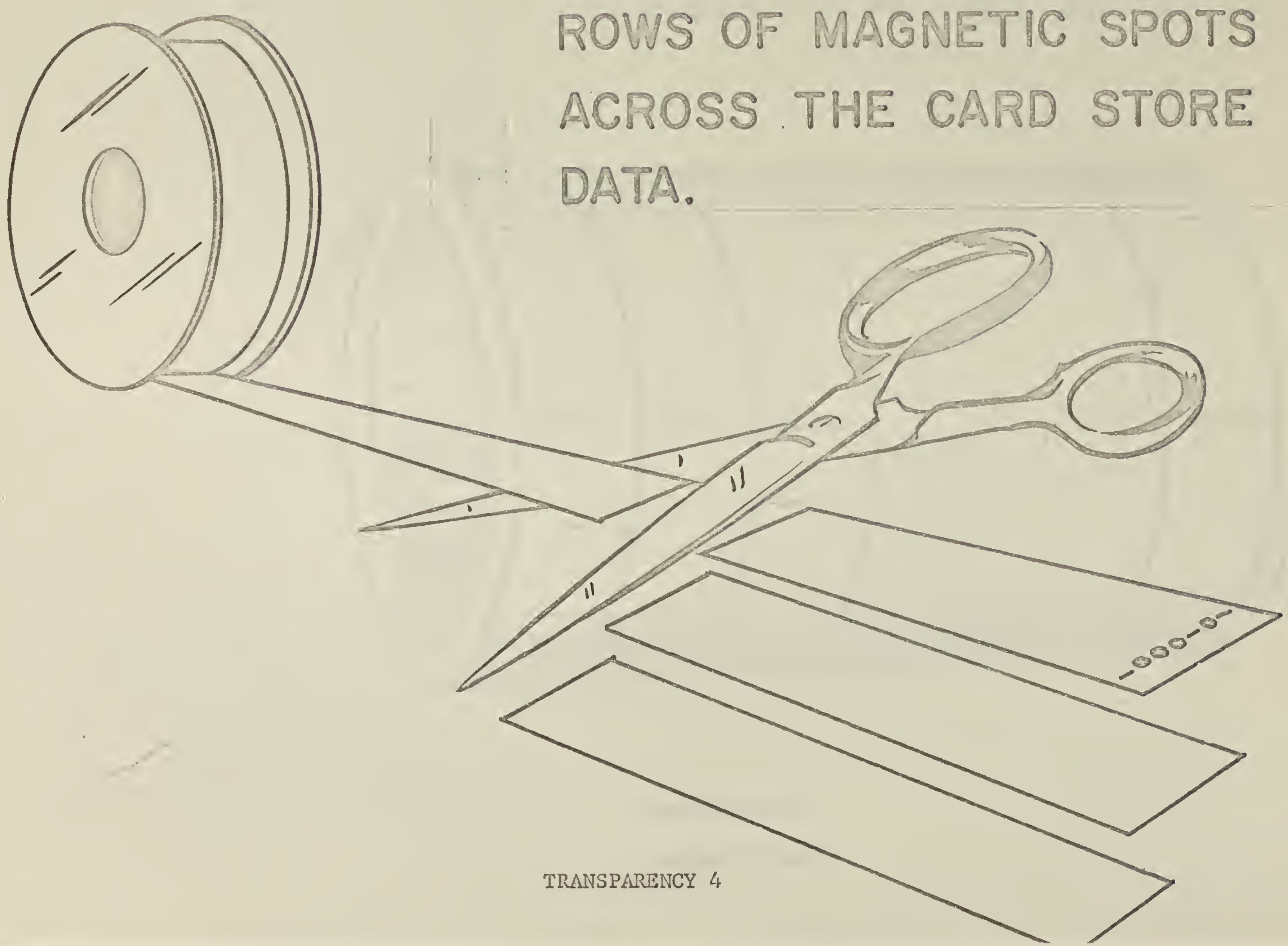
Read / Write Head



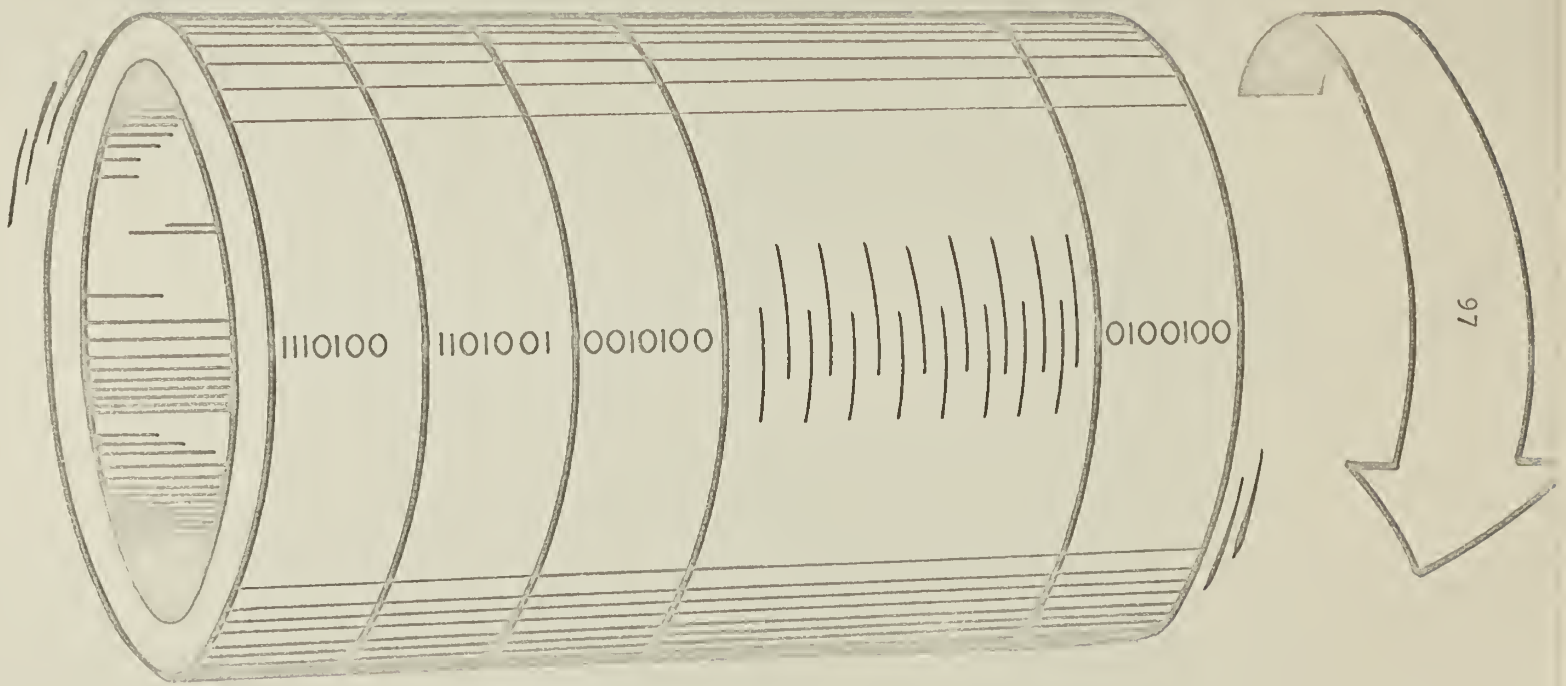
TRANSPARENCY 3

MAGNETIC CARDS

ROWS OF MAGNETIC SPOTS ACROSS THE CARD STORE DATA.

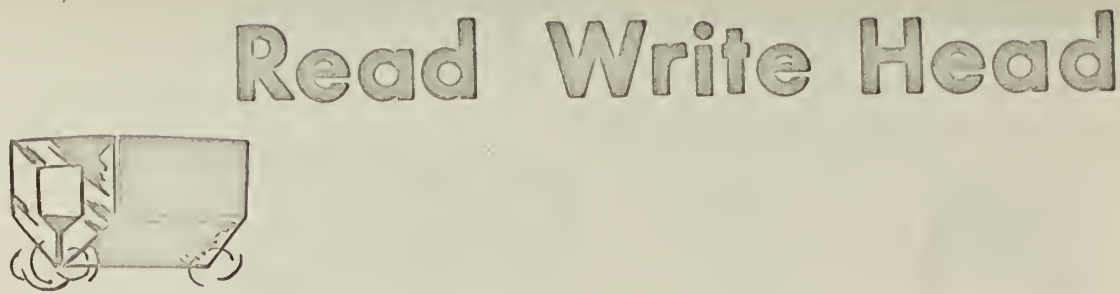


TRANSPARENCY 4



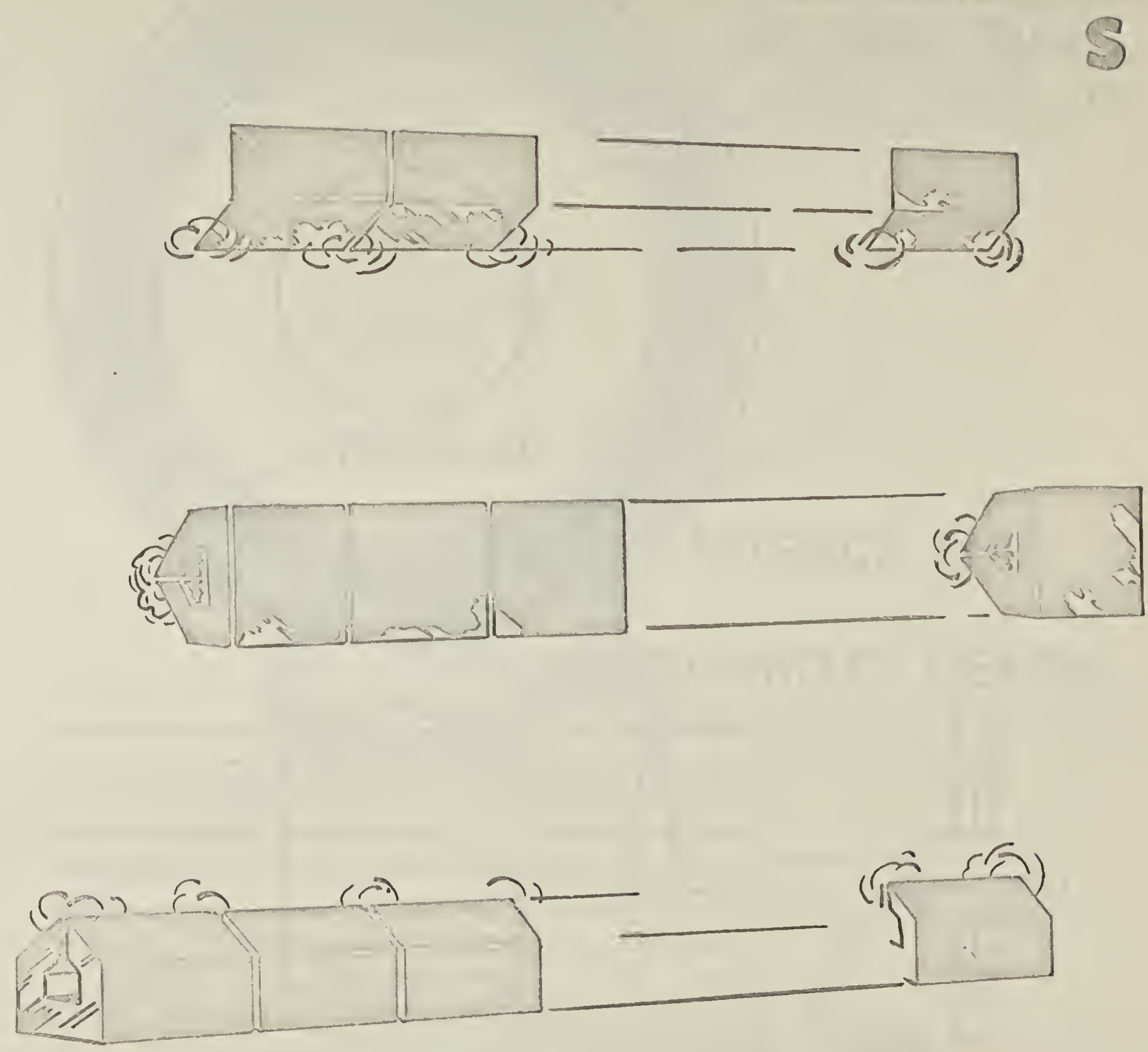
TRANSPARENCY 5

MAGNETIC DRUM



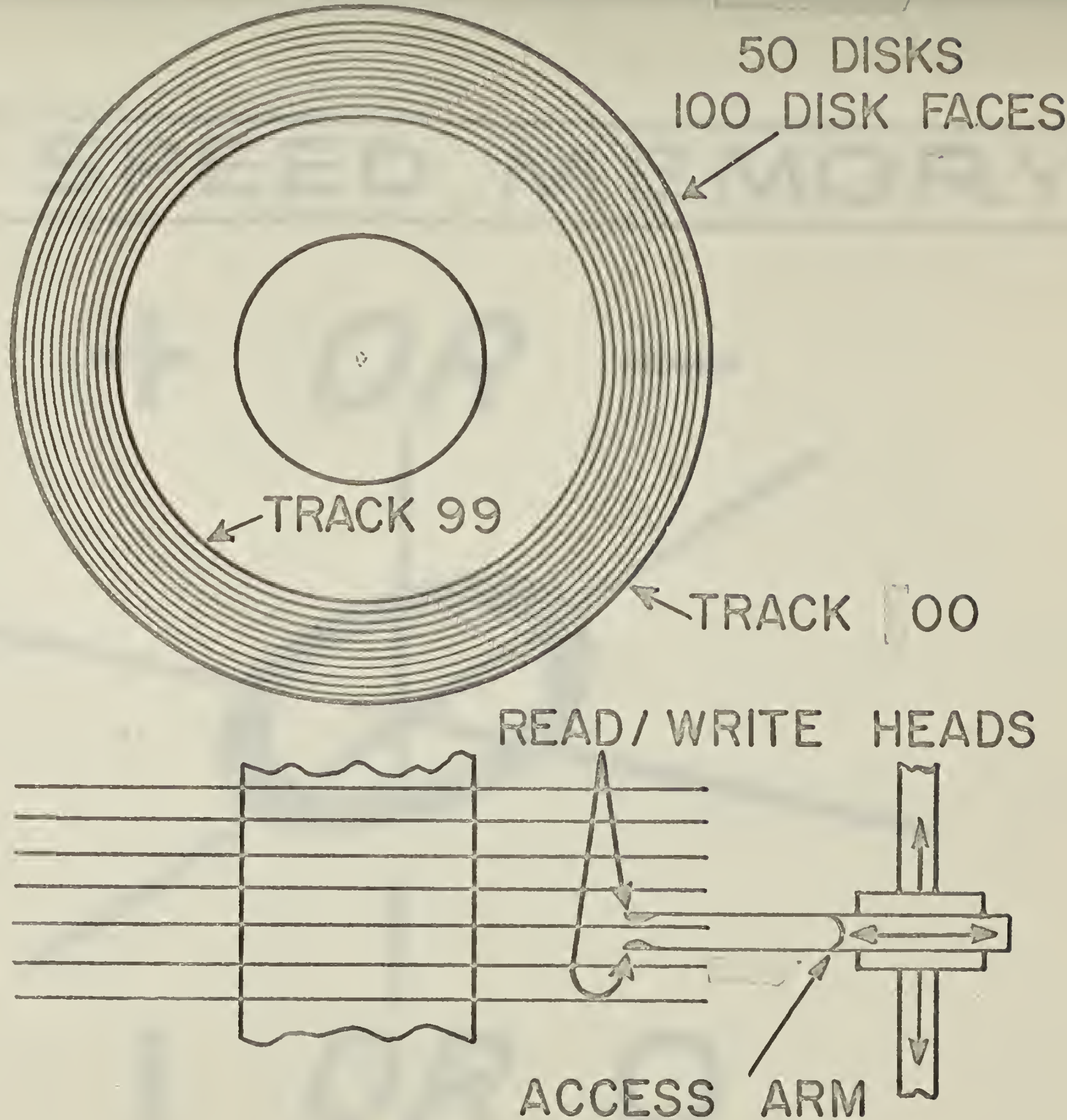
TRANSPARENCY 5

MAGNETIC DRUM



TRANSPARENCY 5

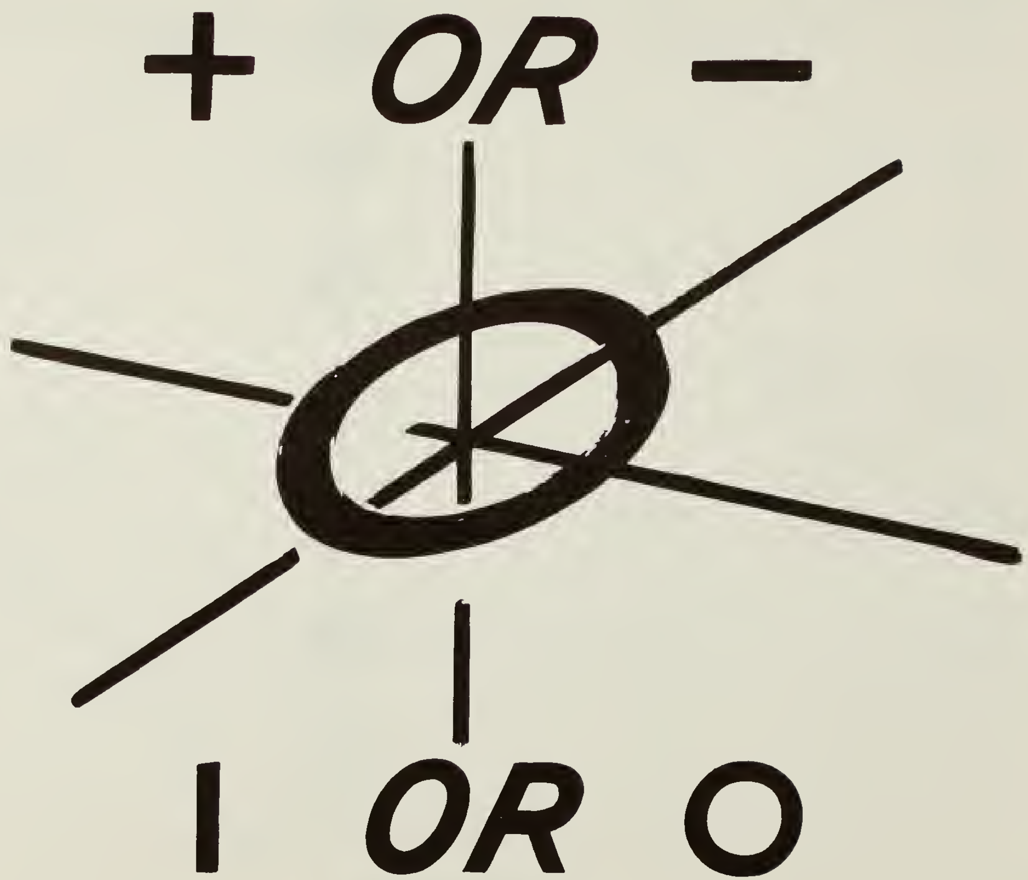
MAGNETIC DRUM



TRANSPARENCY 6

MAGNETIC DISC

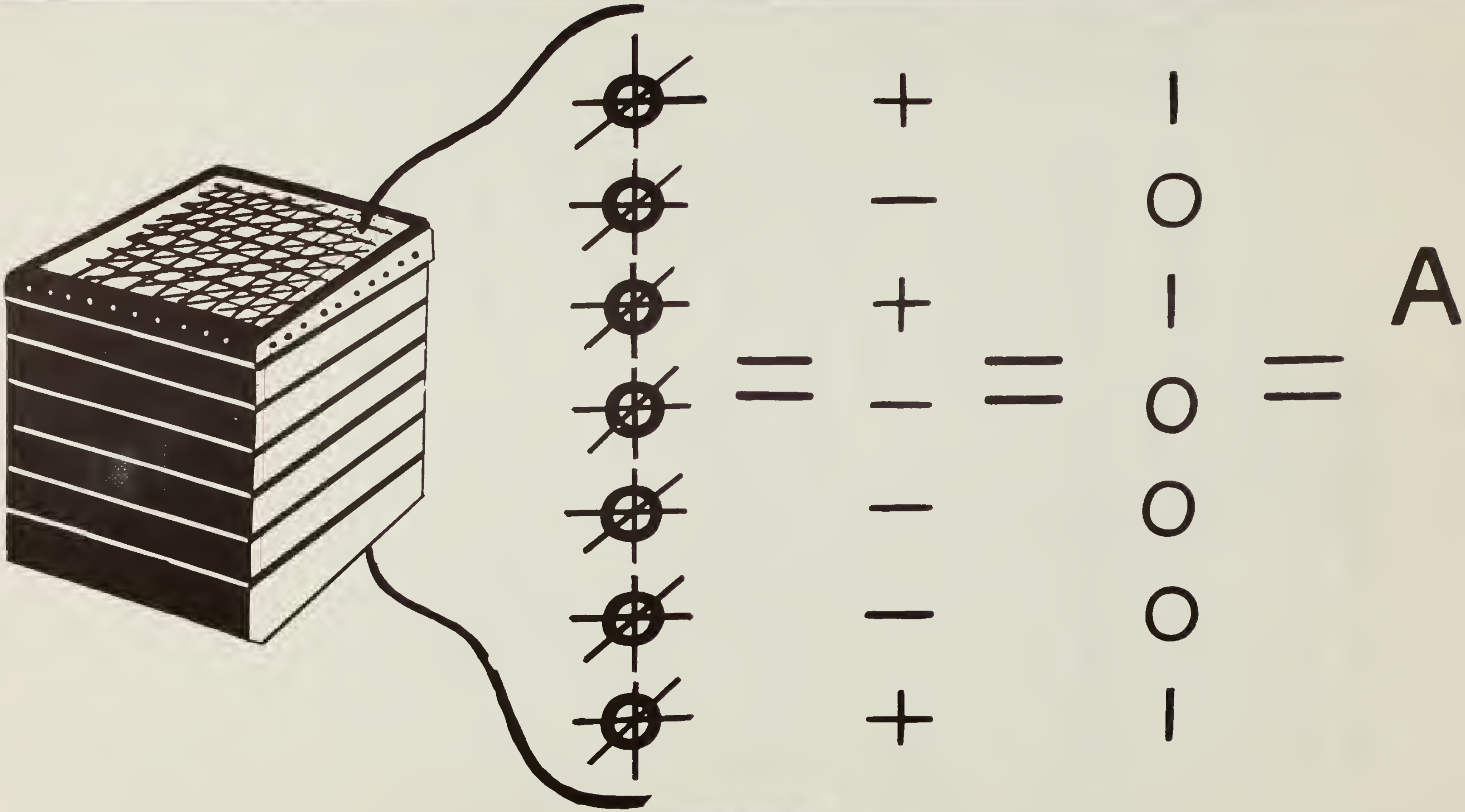
HIGH SPEED MEMORY



TRANSPARENCY 7

MAGNETIC CORE

HIGH SPEED MEMORY



TRANSPARENCY 8

MAGNETIC CORE MATRIX

D -- INPUT/OUTPUT

COURSE: AUTOMATIC DATA PROCESSING

SESSION TITLE: INPUT/OUTPUT

OBJECTIVE: Familiarization with Input/Output Devices and Capabilities

TIME: 45 minutes

TRAINING AIDS REQUIRED: Overhead Projector

Two (2) Transparencies

Film -- "Input-Output" MN8969 (Available from Film Library, MCAS, Cherry Point, N. C.)

35 mm Slide Projector

16 mm Projector

Porto-punch

HANDOUTS: Punched Cards

Paper Tape

Example of Print-Out

REVIEW

This session will be devoted to input-output devices. In the previous session, STORAGE DEVICES, the various means of storing data were introduced. You will recall that we discussed magnetic tape, magnetic cards, magnetic drum, magnetic core, punched-cards and punched paper tape. You will also recall that data is stored in various ways in these devices. In some, it is stored by means of holes, in others by means of magnetic spots or on magnetic cores.

You were shown how magnetic core works -- how a stack of 7 cores were arranged so that a character could be represented. You saw how the stacks of cores were combined into arrays in order that items, records and files might be stored. Finally, you were shown the equipment used for storage and you discussed the characteristics of each type of equipment. You will probably recognize some of these storage devices as input-output devices -- the punched-card for example, or the magnetic tape unit.

We know how we store data, the next question is how do we get data into storage and how do we get data out of storage in usable form.

INPUT/OUTPUT

Input data may be obtained from many sources. However, data, to be useful must be in a form suitable for read-in to a processor.

Perhaps the most familiar method of input for a data processing system is the punched-card. Information in the punched-card is recorded as small rectangular holes punched in specific locations in the card. This data may be read as the card is moved through a card reader.

Reading the card is basically a process of automatically converting data recorded as holes to an electronic language and entering this data into a machine. Cards are used for entering data into a machine, for recording and receiving information from a machine. How do we get holes in the card?

Porto Punch -- by use of a stylus and special perforated cards (Your Card No. 1) the cards may be punched and are ready for machine processing.

Mark Sense -- by use of an electro-graphic pencil (Your Card No. 2) cards may be marked and electrically converted to holes by machine.

(SHOW SLIDE #1)

The most common method of converting source data into punched cards is by using a key punch machine. The operator reads the source document and by depressing keys, converts the information into punched holes. This is very similar to typing on a typewriter. Soon after the cards have been keypunched all cards are processed through the keypunch verifier and will be notched either with an o.k. notch (Pass-Out #2) on the right end of the card, which means the card has been verified as being correct or it will be

notched with an <u>error notch</u> Pass-Out #3) in the card above the column in which the error is first detected. The verifier operator reads the same source document the keypunch operator used.

(SHOW SLIDE #2)

On this side we have two card data transmission devices, also part of the AUTODIN System. Cards may be fed automatically and transmitted or received at the rate of 12 cards per minute. One device is used for transmitting, the other for receiving.

The device in the center (POINT) is called a Circuit Switching Unit. The circuit switching unit is a device which interrogates a computer at a switching center to obtain a line to the station being addressed.

<u>In-Put Device</u>

(SHOW SLIDE #3)

An example of a system which uses data transmission cards and paper tape is the transactor.

The transactor is a sturdy, compact machine which was designed to fit into the general atmosphere of an industrial area. It is connected to the Compiler by a multi-wire cable. Predetermined information is fed into the Transactor by pre-punched business machine cards; fixed information will be wired into the machine plugboard, and variable information will be fed by dials and buttons located on the front panel of the Transactor.

The Compiler is a receiving and recording machine which will receive, validate and record all information received from the Transactors. The messages received by the Compiler will be recorded in the form of punched paper tape. This tape will, in turn, be converted to punched-cards or submitted as direct input into a magnetic tape computer system.

As you can see, this will by-pass many laborious and time consuming manual operations which are necessary under the present system.

(SHOW SLIDE #4)

We can convert the tape into punched cards by use of a card punch machine with a paper tape converter. Now that we have our cards ready to feed the processor, how do we do it? We use a card reader. On-line punched-card readers are used with many processors. The processor can control the card reader and start and stop the reading operation as required. Editing is done under program control to check for double punches, blank columns, and field consistency. Contents of columns can be omitted, altered, or rearranged for transfer into storage. Card readers usually verify accuracy by reading each card twice and comparing for identity, or by reading once and making a hole count for each column for comparison with a second hole-count made at a read-in check station. Conversion between the punched-card code and the code used in the processor is handled by read-in equipment.

(SHOW TRANSPARENCY #1)

Throughout the ADP cycle, great attention is given to checking for accuracy. We must make certain that the data we have transposed is free from human errors and that it is in the correct format. This is called Data Verification. The processor does this: The simplest type of format verification is to ensure that each data field contains the correct kind of characters (numeric alphabetic or both). An alphabetic character in a numerical statement of amount such as $99 A.99 on a check violates the rules of format and is an obvious error which defies further processing. Unless data is keypunched correctly the processor is programmed to reject it.

Next the data is verified to see that all the elements appear to be complete and that the meaning is reasonable. In the second example shown on the screen, the blank space in the figure$-99.99 indicates that there is a character missing, therefore the processor would be programmed to reject this. In the last example $99.999 it is obvious that the figure is unreasonable if our field is limited to $999,99 and the machine would be programmed to reject this data. So much for cards and punched holes.

(SHOW SLIDE #5)

Let's talk about the Magnetic Tape Station. The Magnetic Tape Station can function as both an input or an output device; it transports the magnetic tape and accomplishes the actual reading or writing of information. You remember that during reading or writing, tape is transferred from the file reel past the read-write head to the machine reel.

Several devices are available for transmitting data recorded on magnetic tape. They read data from magnetic tape, transmit data over telephone wires, and record on magnetic tape at the destination. The transmission speed over commercial telephone channels is from 150 to 300 characters per second.

(SHOW SLIDE #6)

Now let's go on to the various printers which are output devices.

Printing devices provide a permanent visual record of data from the Computer System. Speeds of printing vary from 10 to 2,500 characters per second.

As an output unit, the printer receives data in electronic form from the processor. These electronic symbols enter appropriate circuitry and cause printing elements to be actuated. Paper advancing is controlled

by instruction, either directly or in conjunction with a paper tape loop in the printer.

(SHOW SLIDE #7)

The Print Wheel Printer is equipped with 120 or more rotary print wheels. Each print wheel has characters of type including numerals, alphabetical symbols, and special characters. At the time of printing, all of the print wheels are correctly positioned to represent the data to be printed. Printing occurs as a completed line of 120 or more characters.

(SHOW SLIDE #8)

Next is the Matrix-Printer. In the wire matrix printer, each character is printed as a pattern of dots formed by the ends of small wires arranged in a five by seven dot rectangle.

By extending selected wires, the patterns may be arranged in the shape of 47 different characters, including all letters of the alphabet, the digits 0 to 9, and a number of special characters. Selected wires are pressed against an inked fabric ribbon to print the characters on paper.

(SHOW SLIDE #9)

The chain printer is an electromechanical line printer using engraved type. Alphabetic, numeric, and special characters are assembled in a chain. As the chain travels horizontally, each character is printed as it is positioned opposite a hammer that presses the paper against one piece of type in the moving chain. Up to 132 positions may be printed on one line at a speed of <u>600</u> lines per minute.

(SHOW TRANSPARENCY #2)

THE NEXT PRINTING DEVICE IS THE PRINT DRUM.

The Print Drum is a continuously rotating cylinder with raised characters arranged on each band around the cylinder. Every character to be used is located in each bank to provide one printing position. A fast-acting hammer opposite each band of characters strikes the paper against the desired character at the correct time. Printing speeds of 800 to 1,000 lines per minute of 160 characters per line are possible.

(TURN TRANSPARENCY OFF)

NOW FOR THE LAST OF THE PRINTERS.

The typewriter used as an output device is similar to the ones used manually. The major difference is that control of the typewriter and the printing are accomplished automatically as directed by the stored program. Printing speed is about 600 characters per minute; spacing and carriage return are automatic.

(SHOW SLIDE #10)

The Interrogating Typewriter is a remote inquiry station. Such stations enable an inquirer to interrogate the processor and receive an answer without manual intervention. This feature is useful if the processor has a large amount of data readily available in addressable bulk storage and the inquirer can quickly use the output for either decision-making or operating purposes.

(SHOW SLIDE #11)

Output from the computer system is also recorded in cards produced by a card punching machine. The card punch automatically moves blank cards, one at a time, from the card hopper under a punching mechanism that punches data received from storage. After the card is punched, it is moved to a

checking station where the data is read and checked with the information received at the punching station. The card is then moved to the stacker.

Card punching speed varies from 100 to 250 cards per minute.

Several of the devices that we have talked about may have what is called "a buffer storage."

A processor can ordinarily accept and record data in storage faster than input units can supply data. Buffer storage is used to compensate for this difference in operating speeds. A buffer may be a small intermediate storage unit between the input unit and processor storage. In other cases, part of the main storage is used for buffering.

Now let's look at a short film which deals with in-put, processing, and out-put which will show these devices. This will serve as a review of basic in-put and out-put devices. (SHOW FILM)

SUMMARY

The input and output subsystem consists of devices and media for getting data into and usable results out of a processor; it is a critical element of a successful data-processing system. The chain of activities leading up to processor input are (1) recording events, (2) converting data to machine language, (3) verifying the data, (4) transmitting the data and (5) the actual read into the processor.

Input preparation usually starts off-line from the processor because it is a relatively slow process. Devices used for converting data into processable form are paper-tape punches, keypunches, typewriters with connected card punches, and magnetic-tape writers.

Various schemes of sight reading and duplicate operations with comparisons are used for verifying that the original data are correctly converted into processable form.

A variety of data-transmission techniques are available: they are telephone, telegraph and microwave. Media used for data transmission are punched paper tape, punched card and magnetic tape.

Input readers connected directly to the processor range from manual keyboards operating at a speed of a few characters per second to magnetic-tape units as fast as several thousand characters per second. Magnetic tape has the desirable features of high-density storage, rapid data transfer, read-reverse, backspace, reliability and reusability.

A most interesting development in input readers is the use of the interrogating typewriter designed for operations to make file interrogations, read in transactions, and receive immediate replies from the processor.

The most common high-speed output medium is magnetic tape, since it has the same desirable features for writing that it does for reading.

High-speed printers that print 500 to 1000 full lines of alphanumeric characters per minute are now commonplace.

DATA VERIFICATION

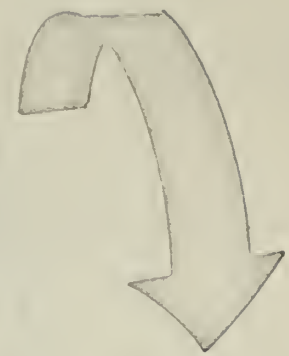
Format . . . .? $99A.99

Complete . .? $ 99.99

Reasonable.? $99,999

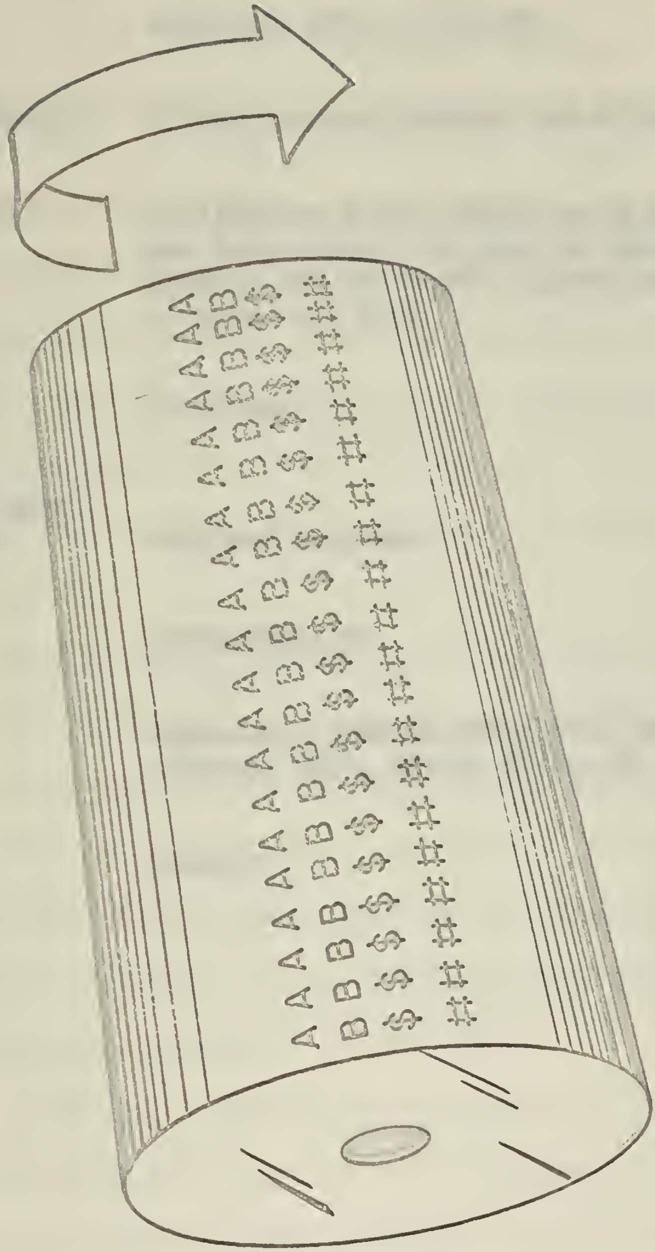
TRANSPARENCY 1

PRINT DRUM



800-1000 LINES PER MIN
160 CHARACTERS PER LINE

TRANSPARENCY 2



TRANSPARENCY 2

E -- STORED PROGRAM CONCEPTS AND SYSTEMS DESIGN

COURSE: AUTOMATIC DATA PROCESSING

SESSION TITLE: STORED PROGRAM CONCEPTS AND SYSTEMS DESIGN

OBJECTIVE: To illustrate what comprises a program, how instructions are used to develop a program and how block diagrams are constructed and used.

TIME: 60 minutes

TRAINING AIDS REQUIRED: Overhead Projector

Transparencies

Magnetized Symbols (Available from Training Aids Library, MCAS, Cherry Point, N. C.)

Pointer

STORED PROGRAM CONCEPTS AND SYSTEMS DESIGN

This class session concerns stored program concepts and systems design. You will learn what comprises a program, how instructions are used to develop a program, and how block diagrams are constructed and used.

(HYPOTHETICAL SITUATION -- ADP EQUIPMENT READY TO GO BUT -- NO PROGRAM)

Before that, however, let us visualize one moment in the very near future when a new computer system is installed. The processor with its "on-line" input and output devices, its external storage devices and its communications units have been checked out by the manufacturer's technicians and the system is ready to go. All that is required is for the power unit to be turned on.

The high speed storage unit is standing by to receive data from input units (magnetic tape and/or punched card readers.) The control unit is ready to direct the flow of data through the system and to initiate the desired arithmetical or logical operations. The arithmetic and logic unit is prepared to perform multiplication, division, addition and subtraction operations as required. It is ready to compare data to determine greater than, less than or equal to relationships.

We know our input devices are ready to move data quickly into the high-speed processor. Our output devices are ready to furnish output in the form we desire. We have established a means to collect data about events as they occur, and we have translated those events into symbols, and we have recorded those symbols as data. We have converted that data into processable form by keypunching and verifying it, or maybe we put it

on magnetic tape, so that it is ready for read-in to our processor, by mailing, by messenger, or by transmitting it over a telephone circuit via the transceiver to the processor. Remember, that if we have transmitted data from one location to another over some form of electrical or electronic circuit such as wire, cable or microwave, it has been at a low, intermediate or high transmission rate. If it was a telegraph circuit, it was a low capacity transmission. The teletypewriter transmits punched paper tape at the rate of six to ten characters per second.

If it was cable transmission utilizing telephone circuits, it had an intermediate capacity of transmission. Data on punched cards can be transmitted and received via the transceiver at the rate of 12 per minute.

If it was a microwave transmission it had a high capacity of transmission up to several thousand characters per second.

If our input-output devices were connected directly to the processor, they were "on-line," and "off-line" if not connected directly to the processor.

Perhaps some of our data came from porto-punch -- a device for manually punching holes in cards by means of a stylus and specially perforated cards.

Perhaps some came from mark sense cards, where a machine senses the presence of a special graphite pencil mark in each card column. Perhaps it came from a paper tape converter, an "off-line" device which converts input data recorded on paper tape into punched-cards.

In any event, our data has ended up at a card reader, an "on-line" device used for read-in of the input data to the processor, or at a magnetic tape unit which is to read-in our data.

On the output end, our printer is ready to provide a permanent visual record of data from the computer system.

It doesn't make any difference whether we have printwheel printer which is equipped with 120 rotary print wheels or a wire matrix printer which records characters as dots to print out 120 characters per line at a rate of 500 to 1000 lines per minute, or a chain printer consisting of a horizontally moving chain of alpha, numeric, and special characters, or a typewriter printer similar to the ones used manually, but printing at a speed of 600 characters per minute, or a print drum printer consisting of a continuously rotating cylinder with raised characters arranged in bands around the rim of the cylinder.

We are ready to print out. Our card punch output devices are connected to and controlled by the processor for "on-line" output and are ready for action. Also at the ready are our interrogating typewriters, a remote "on-line" inquiry device used to obtain information from addressable bulk storage. In short, our ADP system is ready to go. The ribbon cutting official is present with the symbolic shears which will trigger off the power supply to the system. All stations are manned, the speech is made, the ribbon is cut, the power supply is turned on and the console operator pushes the appropriate start button. NOTHING HAPPENS !!!! NO RESPONSE !!!! except that the typewriter at the console is hammering away a message while the operator is frantically pushing the start button.

(SHOW TRANSPARENCY #1)

THE PRINTED MESSAGE?????

"Dear Operator -- I cannot function without the necessary instructions in my internal storage. Please postpone your ribbon cutting ceremony until

<u>such time as you have read-in your required programs</u>."

Of course, this is stretching the imagination quite a bit. This could never happen here. But we chose this method to review our system and to emphasize the fact that the computer acts upon instructions created by man and that man is still the controlling factor in man versus machine. You just don't push a button, whistle, and get your answer. You've got to program your requirements so the machine can understand them. After input data is read-in to the processor, the processor takes over the complete processing and the preparation of results. However, the procedural steps that are to take place within the processor must be defined precisely in terms of operations that the system can perform. Each step must be written as an <u>instruction</u> to the computer. An instruction then is a unit of specific information of an operation located in main storage. In automatic data processing systems, instructions are stored internally in main storage and the system has access to the instructions at electronic speeds. A series of such instructions is called <u>stored programs</u>. This is the distinguishing feature of automatic data processing -- this ability to store instructions and programs internally, and to change them at will without stopping processing.

<u>(PROGRAMS AND EAM SYSTEM)</u>

You see before you an IBM electronic accounting machine panel board. The wored panel board represents a series of instructions to an EAM machine. Back in the early 1950's a systems program consisted of a series of these wired panel boards, one for each EAM machine used in the processing. The wires represent the various instructions to the machine. There were several machines and many, many wired panel boards. For example, the EAM listing

machine or printer would use one panel board for running a personnel listing, another panel board for a supply inventory report, and still another for a fiscal accounting report. So you take each machine (and there are quite a few), add a wired panel board for individual instructions, and you end up with a wired panel board library. Now, in an electronic data processing system these instructions can be stored internally in main storage. No more wires -- no more boards -- more flexibility and far more capacity.

Remember that the computer is directed to perform each of its operations by an instruction -- a unit of specific information located in main storage. This information is interpreted by the processor as an operation to be performed.

You recall that each instruction consists of at least two parts:

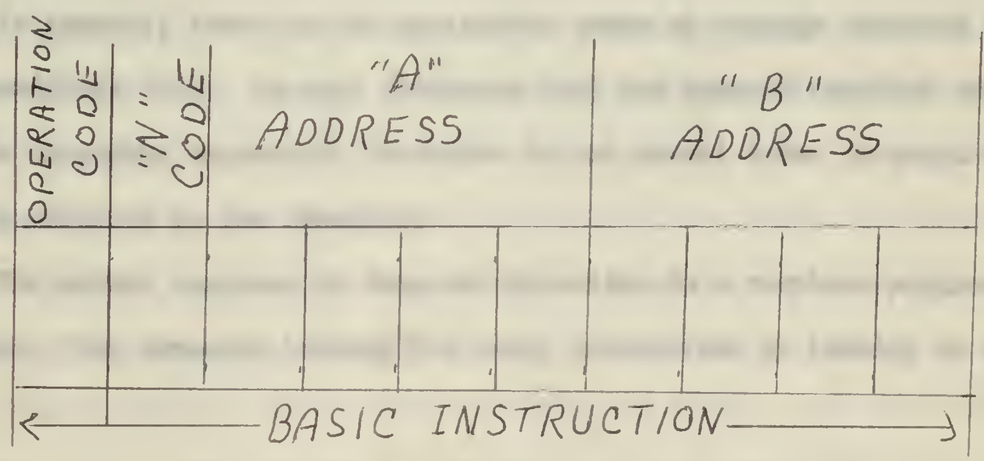
1. An operation part that designates read, write, add, subtract, compare, move data, and so on.

2. An operand that designates the address of the information that is needed for the specified operation.

In some computers, instructions have two address portions.

(THE BASIC INSTRUCTION) (BOARD WORK)

Let's consider a 2 address 10 character instruction.



O = Operation code of one character which identifies the operation of command, e.g., R (READ) -- W (WRITE) -- M (MOVE) -- C (COMPARE) -- T (TRANSFER)

N = Consists of one character to indicate -- a count, e.g., M (MOVE) 5 characters: a symbol, e.g., T (CONDITIONAL TRANSFER OF CONTROL) 1 execute next instruction: an input-output device symbol, e.g., R (READ) 3 = read from TAPE UNIT #3, W (WRITE) 4 = write to TAPE UNIT #4.

The operand further defines the function of the operation. For example:

"A" Address = Consists of four characters and refers to access from a specific location in HSM, e.g., M (MOVE) 4 8000 = move 4 characters from storage located in 8000.

"B" Address = Consists of four characters and refers to destination in storage, e.g., M (MOVE) 4 8000 8600 = move four characters from storage located in 8000 to location 8600.

Because all instructions use the same storage media as data, they must be represented in the same form of coding. Instructions are usually fixed in length.

In general, there are no particular areas of storage reserved for the instructions only. In most instances they are grouped together and placed in ascending sequential locations in the normal order in which they are to be executed by the computer.

The normal sequence of computer operation in a complete program is as follows: The computer locates the first instruction by looking in a

pre-determined location of storage assigned for this program. This first instruction is executed. The computer then locates the next instruction and executes it. This process continues automatically, instruction by instruction, until the program is completed or until the computer is instructed to stop.

How do we get instructions into internal storage?

Using the basic instruction form of 10 characters, the instructions are punched on cards, and the cards are then read-in to the processor or the card reader. Cards are read-in to the processor for internal storage. They may be also recorded on a magnetic tape for faster read-in when required.

(SHOW TRANSPARENCY #2)

Let us look at this transparency of a magnetic core array comparable to the HSM storage unit in the processor and see how the instructions enter internal storage. Remember that each character can represent a numeral, a letter or a symbol. Each character consists of seven bits. The letter A is punched on the cards as an A and upon read-in to the processor is converted to machine language or bits. The bits are translated into electrical pulses which change the polarity of each magnetic core for each bit represented here, by a plus for the one bit and a minus for the zero bit. Thus the transfer into storage is similar for each character of the instruction until the entire instruction is in storage. Each subsequent instruction is transferred in the same manner until the entire program is stored. There are processors today which can transfer a ten character instruction simultaneously or in parallel.

Now we are back to our ribbon cutting ceremony. Once the instructions are stored, input data can now be processed. Data is supplied via an input

device and is transferred into Main Storage in the same manner.

Now let's turn our attention to the program -- that is, the assembly of instructions in logical order.

The first step in program preparation is a complete analysis of the machine method and the procedure, either existing or proposed. This analysis is normally accomplished by developing flow charts and block diagrams, because most data processing applications involve a large number of alternatives, choices and exceptions.

It is difficult to state these possibilities verbally or in written form. Thus, the systems analyst finds uses for many types of pictorial representations, including form layouts, control panel diagrams, manpower loading charts, and so on. The two representations to be discussed here are the flow chart and the block diagram.

A flow chart, is a graphic representation of the data processing system in which information from source documents is converted to final documents. A flow chart provides a picture of the data processing application from the standpoint of what is to be accomplished. Such a picture gives primary emphasis to the documents involved and secondary emphasis to the work stations through which they pass. A block diagram is a graphic representation of the procedures by which data are processed within a system. In this picture the emphasis is on the operations and decisions necessary to complete the process.

To summarize: a flow chart shows what job is to be done; a block diagram shows how a job is done.

The analyst needs a method to communicate so standardized symbols have been developed to use in block diagramming. Through the use of templates, symbols are drawn to represent clerical functions, machines and

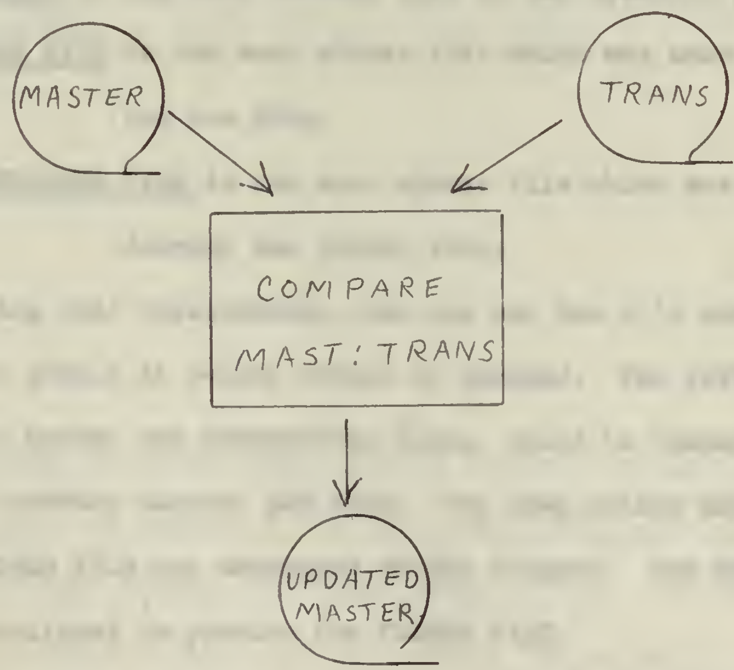
functions, data processing systems and functions, and types of documents.

We will be primarily concerned in this class with block diagramming, although the 2 terms -- flow charting and B.D. -- are used interchangeably in practice. If you will refer to your handout, we will learn the most commonly used standard symbols by which block diagramming is accomplished.

(DESCRIBE THE SYMBOLS)

Let's take a problem -- let's say we had a file of items identified by stock number, and we had certain transactions which we wanted to add to the file. These files are on tape, and we have master tape which contains all items, and a transaction tape which contains all transactions. We want to update this file to show all transactions.

(SHOW TRANSPARENCY #3)



Now that the class is familiar with some of the symbols used in block diagramming, let's see how these symbols are used in developing a

block diagram flow. We know that a block diagram is a bird's eye of step-by-step operations of a procedure, so let's take a hypothetical problem and flow the desired operations in the sequence that they occur.

We have a file of items identified by stock number. This file is the master file of all items of inventory. The other file is a transaction file which we want to add to the master file to reflect the current status of each item of inventory for which there is a transaction. The stock numbers of each file are compared and the updated record is written on a third tape. Why a third tape? There are two very good reasons.

FIRST REASON -- In an ADPS we have safeguarding system for all file maintenance. It's called the son-father-grandfather concept and when related to systems files it means that the:

-- SON FILE is the most current file in any specific program.

-- FATHER FILE is the next oldest file which was used to develop the son file.

-- GRANDFATHER FILE is the next oldest file which was used to develop the father file.

Now, using this relationship, you can see how it's easy to develop the current file should it become erased or damaged. The FATHER FILES, in this case the master and transaction files, could be compared together again to produce another SON FILE. The same action would apply in event the FATHER FILE was destroyed and/or damaged. The GRANDFATHER FILES would be utilized to produce the FATHER FILE.

SECOND REASON -- Recall from your previous session on magnetic tape storage what happens to the old information when new information is being written on the tape. The old information is erased by electrical impulses as the

new information is being written. This erasure would destroy what previous information was stored on the tape and would be lost forever. So you now see why it's necessary to write-in the updated record on the third tape. Now that we have this point cleared, let's continue with our hypothetical problem.

Let us see what a record is. Our records on each tape will consist of 14 characters, where the stock number is represented by four characters, the name is represented by five characters and the quantity is represented by five characters, e.g.,

(BOARD WORK)

STOCK NO.	NAME	QUANTITY
4	5	5

Let's review the steps which make up a file. We know that it takes seven bits to form a character, it takes a group of related characters to form an item, it takes a group of related items to form a record, and it takes a group of related records to form a file. Right now we will take the first record from our transaction file and it may read like this.

	STOCK NUMBER	NAME	QUANTITY
	4	5	5
TRANSACTION	0001	CLOCK	00001

Our first record from our master file may read like this.

MASTER	0001	CLOCK	00500

Now when we compared these two records, our updated master record would read like this.

NEW MASTER	0001	CLOCK	00501

This is exactly what we want to block diagram. So let's use our symbols to draw the step-by-step operation flow as they must occur in order to provide detailed instructions to the processor.

<u>HYPOTHETICAL PROBLEM</u> (SHOW TRANSPARENCY #4)

Our Hypothetical Problem is to update a master stock record file. We have been limited to 100 positions of core storage or memory located in our make-believe computer. Our magnetic core array will consist of 7 matrices, each matrix having a 10 bit by 10 bit core storage capacity.

Our input data will be contained on a master magnetic tape file and a transaction magnetic tape file. Each file is in a sequential order, that is, we start with the lowest stock number and continue progressively to the next highest stock number until the end of the file. We know that the transaction file has been sorted, that it contains all the receipts for one day's transactions, and that the receipts have been consolidated to reflect the total day's receipts for each stock number. Since our transactions are receipts, we know that our only arithmetical computation will be to add each transaction receipt to each comparable master inventory record.

Each record in the file consists of 14 characters, and we have one record per block recorded on the respective magnetic tape file. Each record is divided to allow for 4 characters to represent the stock number, to allow for 5 characters to represent the name, and to allow for 5 characters to represent the quantity. We will leave our Hypothetical Problem on the screen while we build our block diagram. We may need to refer to it.

(BUILD BLOCK DIAGRAM ON BLACKBOARD USING MAGNETIZED SYMBOLS)

Remember that each block diagram must have a start and that our Hypothetical Problem concerns one of the many operational phases of the program, STOCK RECORDS MAINTENANCE. Therefore, the start symbol is reflected in the beginning of the program and we will not be concerned with it at this point of the operation flow. As I build this block diagram on the board, I recommend that you draw your own flow of the operations using the same symbols as I do on the board. Save this block diagram for use in the next session of the Hypothetical Computer.

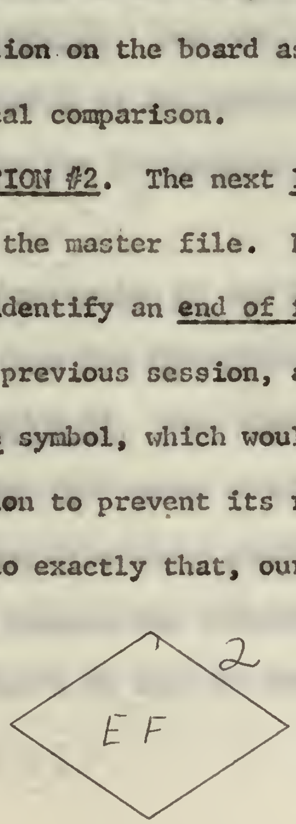
Our Hypothetical Problem involves the comparison of stock numbers, whether the stock numbers are equal to each other, whether one stock number is greater than the other stock number, or whether one stock number is less than the stock number. In building our block diagram, we must recognize that these conditions will exist and that we will have to provide for an operational flow to take care of each condition. We also know that the block diagram should be in a straight flow in the sequence of the operations as they occur. With this in mind, let us flow chart one of the compare conditions which is most likely to provide us quickly with a bird's eye view of the majority of the operations. If we start with the condition

that the stock numbers to be compared are equal, then we will be able to record the flow of the main operations. Since these operations will become instructions to the computer, let us number each operation in the sequence that it occurs in order to maintain identity and continuity in the operational flow.

OPERATION #1. Our first operation then will be to read the first record from our transaction file. Draw the symbol for a magnetic tape

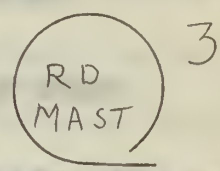
(PLACE MAGNETIZED SYMBOL ON BOARD), insert the abbreviations RD for read, TRAN for transaction, and identify the operation by placing the numeral 1 alongside the symbol. These abbreviations will help us identify the specific operation in the diagram. I'll record the stock number of the first transaction on the board as TRAN 0001 to facilitate our understanding of the numerical comparison.

OPERATION #2. The next logical operation would likely be to read a record from the master file. However, this is wrong for we must tell the processor to identify an end of file symbol on the tape. If you'll remember from a previous session, a magnetic file tape would contain such an end-of-file symbol, which would cause the processor to stop the tape upon recognition to prevent its running off the spool. Since we want the processor to do exactly that, our second operation would be a diamond-shaped symbol (PLACE MAGNETIZED SYMBOL ON BOARD)



to represent a decision that the processor must make -- is this or is this not the End-Of-File? Enter the abbreviation EF re represent End-Of-File and identify the operation as number 2. Draw a line connecting Operation #1 with Operation #2 and insert the arrowhead  to indicate diagram flow. This is not the end-of-file for our purposes right now since we are trying to establish the main flow of our diagram.

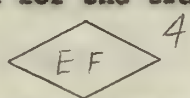
OPERATION #3. Now our next operation is read the first record from our master file. Draw the symbol for a magnetic tape



(PLACE MAGNETIZED SYMBOL ON BOARD)

insert the abbreviations RD for READ, TRAN for TRANSACTION, identify the operation as number 3, and connect operation 2 with operation 3 with a directional flow line. Now, I'll record the first stock number from our master file on the board as MAST 0001.

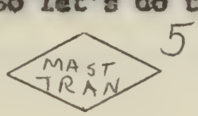
OPERATION #4. Our next operation would be what? To recognize an End-Of-File symbol on the magnetic tape. Since this is the same decision that we had for the transaction tape, let's draw the decision symbol,



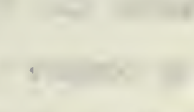
(PLACE MAGNETIZED SYMBOL ON BOARD)

insert the abbreviation EF for End-Of-File, identify the operation as number 4 and draw the directional flow line from operation 3 to operation 4.

OPERATION #5. Now, what have we got? We have the first record from our transaction file whose stock number is 0001 and we have the first record from our master file whose stock number is 0001. We now have to direct the processor to compare the transaction stock number with the master stock number. So let's do this by drawing the decision symbol,

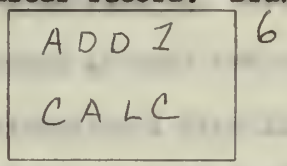


(PLACE MAGNETIZED SYMBOL ON BOARD)



insert the abbreviations MAST:TRAN, identify the operation as number 5 and draw the directional flow line from Operation 4 to Operation 5.

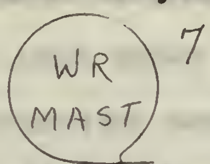
OPERATION #6. Since the main flow of our diagram concerns equality of numbers, and our first master stock number is equal to the first transaction number, the processor must be directed to perform one calculation of addition. Add the quantity from the transaction record to the quantity on the master record. Draw your symbol for computer operation



(PLACE MAGNETIZED SYMBOL ON BOARD)

insert the abbreviations ADD 1 CALC to represent ADD ONE CALCULATION, identify the operation as number 6, and draw your directional flow line from Operation 5 to Operation 6.

OPERATION #7. Now that we have provided for an addition of quantities, our next operation is to record the sum, the first stock number and name to the new master file which will be our updated master file. Let's do that. Draw the symbol for a magnetic tape



(PLACE MAGNETIZED SYMBOL ON BOARD)

insert the abbreviations WR MAST for WRITE MASTER, identify the operation as number 7, and draw the directional flow line from operation number 6 to operation number 7.

DISCUSSION. What we have here is a block diagram of the computer operations involved in processing one record from each input file to obtain an updated master file when the record identification is equal.

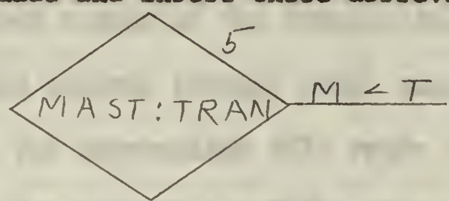
However, without any other flow lines and operations, the computer would not be able to perform the next operation. It must be instructed to perform the subsequent operations even if they repeat themselves. It would be very time consuming to draw another diagram with the same operations. So, in order that the computer can continue processing, we'll draw the next operation.

OPERATION #8. After writing the updated master, the processor is now directed to read the next record from the transaction tape. So let's draw a directional flow line from Operation Number 7 to Operation Number 1 and identify it as Operation Number 8. We have formed a loop in the processing flow and as long as the record from the transaction tape is equal to the record from the master tape the processor can perform these operations without stopping.

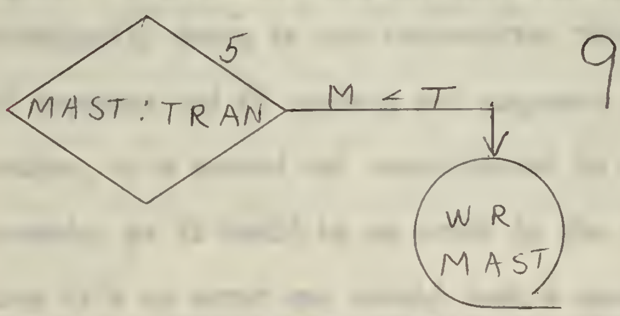
DISCUSSION. What are we going to do for those odd ball situations when the records are not equal? Remember, I told you that we will have to consider these conditions of inequalities. Let's tackle a specific case where the stock number of the master record is less than the stock number of the transaction record. For example, the first 10 records from each file were equal, that is, the stock numbers were in a consecutive sequence and the processor was able to perform one operation after the other and update the master file. Let us say that the stock number of the next record read-in from our transaction tape was TRAN 0012 and that the stock number of the next record read-in from our master tape was MAST 0011. In this case the master stock number is less than the stock number from our transaction record. What does this mean? It means that we've come across

a record in our master file for which there was no transaction recorded; and since this is a condition which will occur whenever the master stock record file is updated, we must make provisions to take care of this situation. The unmatched record on the master file is part of the stock inventory and it cannot be erased and must be recorded on the updated master file. So let's draw the necessary operations and flow lines to do just this.

<u>OPERATION #9</u>. Up to this point, we have read-in one record from the transaction file whose stock number is 0012; this was not the End-Of-File and we have read-in one record from our master file whose stock number is <u>0011</u>; this was not the End-Of-File. The stock numbers are compared and the master stock number 0011 is less than the transaction stock number 0012. Let's draw a flow line from operation number 5 where this comparison is made and insert these abbreviations and symbol



which means that the master file record is less than the transaction file record. Draw a flow line downward and draw the symbol for a magnetic tape



(PLACE MAGNETIZED SYMBOL ON BOARD)

insert the abbreviations <u>WR MAST</u> for WRITE MASTER and identify the operation as number 9.

OPERATION #10. Up to this point, we have recorded the master record, stock number 0011, on the updated master file. We still have our transaction record, stock number 0012, to be compared with the next record of our master file. To direct the processor to the next operation, "read next record from master file," -- what must be done? We draw an operational flow line from operation 9, "write updated master file," to operation number 3, "read record from master file," and identify this operation as number 10.

DISCUSSION. Now, we have two operation loops working for us. One loop to process records whose stock numbers equal each other. The other loop to process records where the master stock number is less than the transaction stock number. Here we have a continuity of operations evolving about an equal to condition and a less than condition. But how about the greater than condition when the stock number of the master record is greater than the stock number of the transaction record? Let us assume that we have compared records through stock numbers 0025 and that the next record read-in from the transaction file reads stock number 0020 and the next record read-in from the master file reads stock number 0026. Here, we have an unequal comparison which indicates to us that something is drastically wrong in our transaction file. The record is obviously out of sequence and if read-in and compared previously, it could be a duplication, or a record not consolidated in with the other 0020 transaction records, or it could be an error in the stock number. Whatever it is, we know it's an error and should such a condition occur during processing, we want the computer to stop operating, if for no other purpose than to accelerate the completion of this HYPOTHETICAL PROBLEM.

OPERATION #11. Let's go to our flow diagram, to Operation #5, where the compare decision is made between the stock number in the record of the master file and the stock number in the record of the transaction file. The decision is that the stock number of the master record is greater than the stock number of the transaction record. Draw a right to left flow line from the decision symbol and insert the characters M < T above it, e.g., M < T 

5. Draw another flow line downward and draw the stop symbol 

11. Insert the word STOP, identify the operation as number 11 and draw the arrowhead symbol to the end of the flow line at STOP.

DISCUSSION. There is yet another condition when it occurs that we want the processor to STOP. It's the End-Of-File symbol on the magnetic file tape. Draw the right to left and downward flow lines to the STOP symbol. We have completed our block diagram and have before us a bird's eye view of the step-by-step operations to solve our HYPOTHETICAL PROBLEM to update the master stock record file. Each numbered operation is considered an instruction to the processor and the total of eleven (11) instructions represents this portion of the program which will be read-in to the processor. The instructions are handwritten on a programmer's prescribed format from which cards are keypunched and verified. The cards are then put into the card reader which reads in the instructions to the processor for transfer into high speed memory storage. Are there any questions?

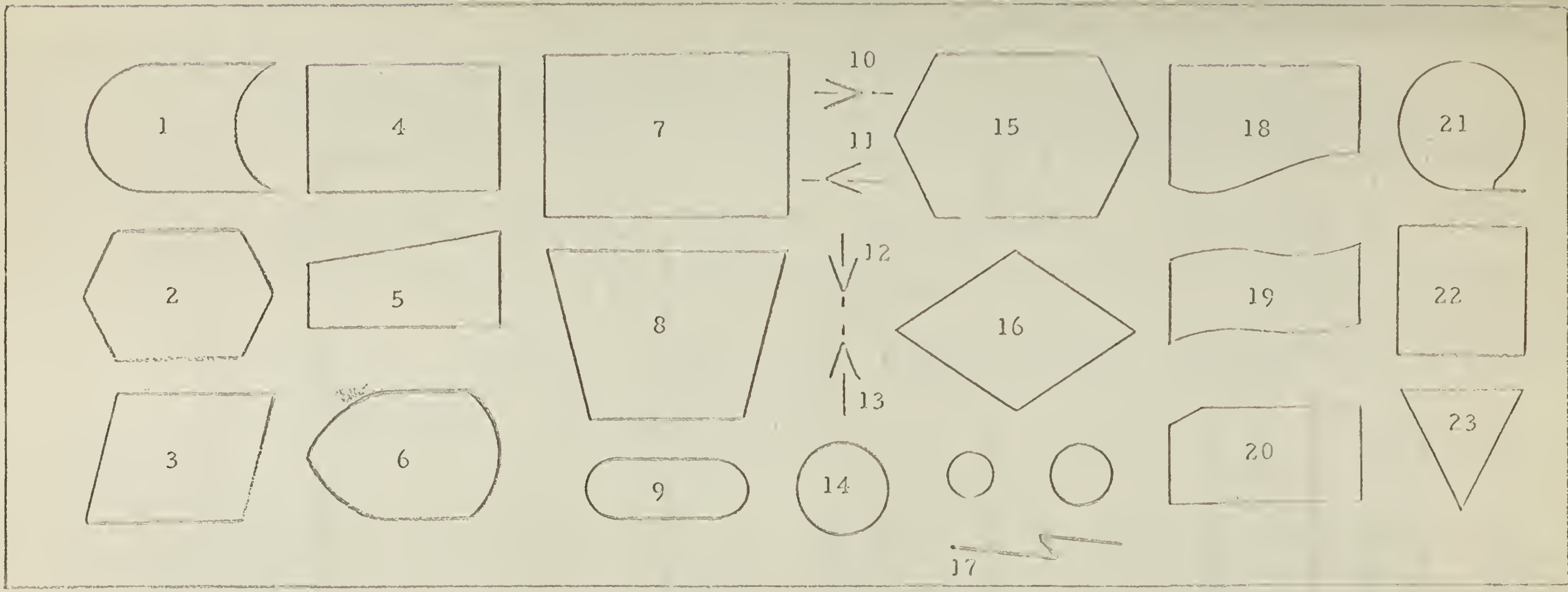
SUMMARY. We have learned that the capability of storing programs internally is a distinguishing feature of an automatic data processing system. Programs consist of instructions which are units of specific information of an operation and are located in main storage in the processor. They are grouped together and transferred into main storage in the sequence that they are to be executed by the processor. The processor selects and executes these instructions in the sequence of the operations that are to be performed. The order of execution may be varied by special instructions in the program to take care of pre-determined condition of data or device within the system.

An instruction consists of two (2) parts. The first part is the operating code which identifies the operation to be performed in terms of a command, e.g., R (READ), W (WRITE), M (MOVE). The second part is the operand which designates the address of the data that is needed for the specified operation. The basic instruction for the RCA 3301 processor consists of ten (10) characters. The left most character of the basic instruction is reserved for the operating code. The next character represents the "N" Code which is used as a supplement to the operating code to indicate a count, a symbol, or an input-output device. The next four (4) characters are assigned to the "A" ADDRESS which refers to access from a specific location in main storage. The last four (4) characters are assigned to the "B" ADDRESS which refers to a specific destination in main storage. The number of addresses to a basic instruction and the number of characters to a basic instruction depend on the capability and design of the automatic data processing system used. Some systems [illegible]. It [illegible] that the [illegible] of [illegible] [illegible] [illegible] to a [illegible] [illegible] [illegible] [illegible] [illegible] system. [illegible] [illegible] [illegible] [illegible] the [illegible] of [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] in the [illegible]. [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] to the [illegible] that [illegible] [illegible] by the [illegible]. The [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] of the [illegible] [illegible] [illegible] [illegible] [illegible]. The [illegible] of [illegible] is [illegible] by [illegible] [illegible] [illegible] the [illegible] [illegible] [illegible] [illegible] [illegible] of data or service within the system.

[illegible] [illegible] [illegible] [illegible] [illegible]. The first part of the [illegible] [illegible] [illegible] the [illegible] to [illegible] [illegible] in terms of a [illegible], e.g., a [illegible], a [illegible], a [illegible]). The second part is the [illegible] which [illegible] the [illegible] of the [illegible] [illegible] [illegible] the [illegible] [illegible]. The [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] of the [illegible] [illegible]. [illegible] [illegible] [illegible] [illegible] [illegible] the [illegible] [illegible]. The [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] of the [illegible] [illegible] [illegible] [illegible] [illegible] [illegible]. The [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible]. The [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible]. The [illegible] [illegible] [illegible] and the [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] and [illegible] of the [illegible] [illegible] [illegible] [illegible]. [illegible] utilize a single address instruction while multiple address computers may use as many as four addresses for each instruction. Instructions are transferred to main storage either "serially" (character by character sequence) or in parallel (the entire instruction at each time cycle of the processor.)

Programs must be charted to facilitate their assembly in terms of instructions or operations. Standardized symbols are used as an aid in systems design and operation flow. These symbols may represent clerical functions, machines and related functions, data processing systems and functions, and types of documents. The pictorial representation developed through use of these symbols result in a flow chart or block diagram. A flow chart shows what job is to be done. A block diagram shows how the job is done. A block diagram shows the flow of the operations in the sequence that they are performed. Each operation is translated by the programmer as an instruction to the processor and is handwritten on prescribed program format for conversion into punched-cards by keypunch and key verify operations. The entire program, composed of these instructions grouped in the sequence of the operations that are to be performed, is read-in to the computer in main storage location and becomes an internally stored program.

Flowchart Symbols Template



1	On-Line Storage	12	Top to bottom
2	Predefined Computer Operation	13	Bottom to top
3	Manual Operation	14	Connector
4	Computer Operation	15	Predefined Computer Operation
5	Manual Input	16	Decision Symbol
6	Display	17	Communications Symbol
7	Computer Operation	18	Document Symbol
8	Input/Output Symbol	19	Punched Tape
9	Start/Stop/Exit Symbol	20	Punched Card
10	Left to right	21	Magnetic Tape
11	Right to left	22	Auxillary Operation
		23	Offline Storage

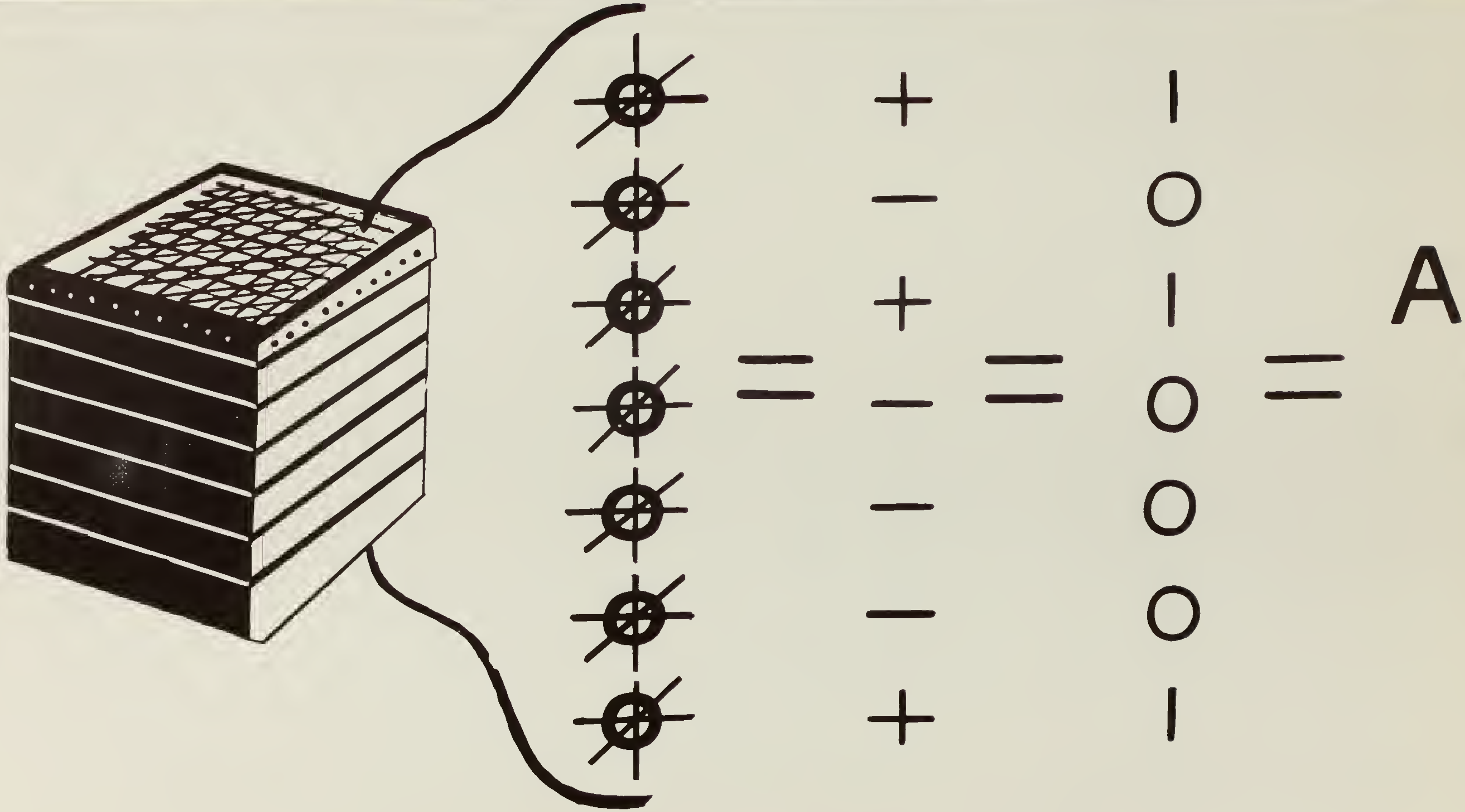
HANDOUT

DEAR OPERATOR,
I CANNOT FUNCTION WITHOUT THE NECESSARY INSTRUCTIONS IN MY INTERNAL STORAGE. PLEASE POSTPONE YOUR RIBBON-CUTTING CEREMONY UNTIL SUCH TIME AS YOU HAVE READ-IN YOUR REQUIRED PROGRAMS.

TRANSPARENCY 1

PRINTED MESSAGE

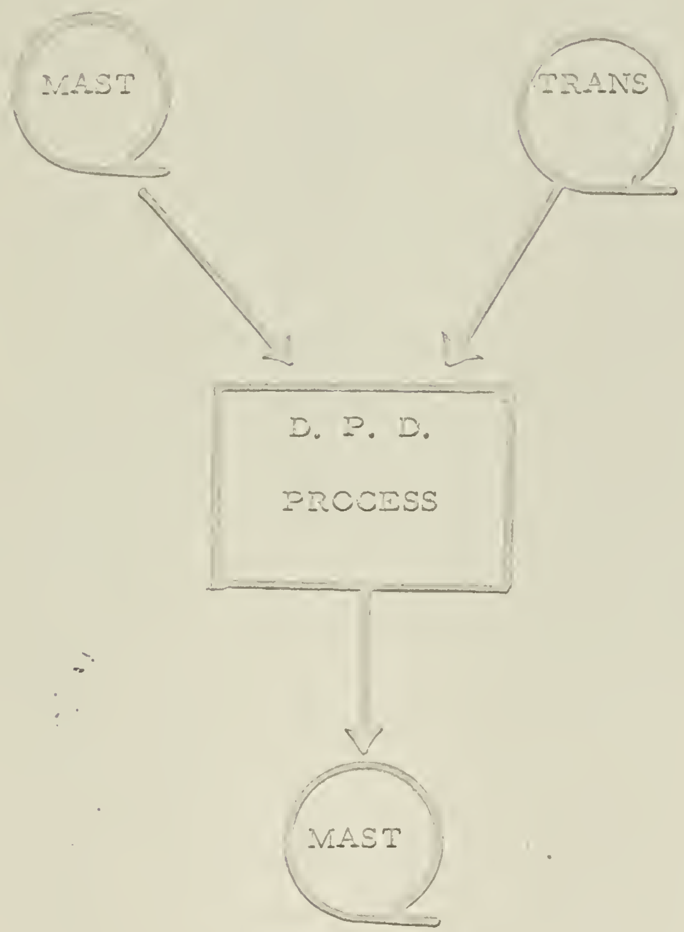
HIGH SPEED MEMORY



TRANSPARENCY 2

MAGNETIC CORE MATRIX

1. UPDATE MASTER TAPE FILE FROM TRANSACTION TAPE -WRITING OUT ON NEW TAPE



TRANSPARENCY 3

FLOW CHART OF PROBLEM

Problem for a 100 position Hypothetical Computer

PROBLEM: Update of Master Stock Record File

INPUT - Master File

Transaction File (receipts only)

Sequential ORDER

14 Characters per Record

1 Record per Block

4	5	5
stock no.	descript.	qty.

TRANSPARENCY 4

PROBLEM FOR HYPOTHETICAL COMPUTER

F -- PROGRAMMING

COURSE:	AUTOMATIC DATA PROCESSING
SESSION TITLE:	PROGRAMMING
OBJECTIVE:	To describe procedure for writing a Program.
TIME:	60 minutes
TRAINING AIDS REQUIRED:	Overhead Projector
	Nine Transparencies
	Pointer
	Four (4) Handouts
	Large Program Record (Available from Training Aids Library, MCAS, Cherry Point, N. C.

PROGRAMMING

This session, like most of the others, will take you back for a short refresher course. By this time, all of you are well acquainted with "high speed memory" and its make-up. It is well to look once again at this important element before going further.

(SHOW TRANSPARENCY #1)

This is one position of core storage. Note the wire extending north and south, and the wire extending east and west -- each carries enough current to create 1/2 the amount needed to activate the core. When both wires are activated at the same time, the sensory wire can then read out the information being held in this storage area. (DRAW PICTURE ON BLACKBOARD OF SEVERAL CORES WITH LINES ACTIVATING ONLY ONE.) Uou will notice the diagram indicates (+ or -) (1 or Ø). The core is in an activated condition when a " + " or " 1 " is present, and in a dormant condition when a " Ø " or " - " is present.

(SHOW TRANSPARENCY #2)

Here you have represented several layers or planes or core. You will note that this represents one hundred positions of core storage -- Count the data -- also, note there are seven layers -- remember, seven bits (binary digits) are required for each character. Take a look at your punched-card -- it is in the Hollerith code. When they are read, an encoder converts this code into binary so that it is usable, and then if a punched-card output is desired, the binary is translated back to the Hollerith. Magnetic tape is written in binary -- so there is no translation necessary. Here is a transparency showing the coded binary form of several

standard characters.

(SHOW TRANSPARENCY #3)

Notice all are combinations of 1 and Ø -- no two are identical.

When we write our program or instructions for our hypothetical computer we will be using 100 positions of core -- so keep the transparency that showed the seven planes of 100 positions of core storage in mind as we go along.

We also need to take a look at the organization of data and the computer instruction format.

(SHOW TRANSPARENCY #4)

Take another look at your 80 column card. What do you see in column 1? A Ø? That is a character. Any letter from A to Z, any number from Ø to 9, or any one of 28 special symbols (%, ?, *) are recognizable characters to most computers. An item is a group of related characters. How many of you are familiar with a labor distribution card. Every person, military or civilian, has a number -- either man number or serial number. The numerical characters that compose these identification numbers are called items. Each one is a unique arrangement to identify one, and one, only, person. A record is a group of related items -- a labor distribution card is a record -- it tells job, rate, hours, etc., all relating to one person's work. The stock status balance card is a record. Your paycheck is a record. A file is a group of related records. Information contained on tape is a record form also.

(SHOW TRANSPARENCY #5)

This is a general representation of the organization of records on magnetic tape. The first item is a beginning of tape label, which contains

the file number and file name, followed by a gap. A gap is just "an area with no writing on it." This happens because the tape station cannot instantaneously achieve maximum speed which is necessary before reading and writing occurs. Next is the tape label which may contain the file number, file name, creation date, retention date and any other desired information followed by a gap, then a record, and a gap. This record is like the records we will use later in the programming problem. Records and gaps are alternated until the end of file indicator is reached. This just signifies the end of file and is followed by a gap and another label. This end-of-file label contains the file number, a record count, a block count, and a hash total (control count). A gap and another end-of-file label generally follow. If this is the end of all data on the tape a gap and end of data label will follow, however, if there is more information it will be recorded as above before the end of data label is recorded. The last item on the tape is an end of tape, window or indicator. This is a mechanical device to signify the physical end of the tape. All of the foregoing are built in safety devices.

The machine instruction format is next.

(SHOW TRANSPARENCY #6)

Please keep in mind that the instruction is vital in that it controls the computer. As you have been told previously, the computer is just a big, flexible idiot with an amazing ability to follow directions quickly. If the human writing the instructions gives a command to the printer to space up until a certain indicator is sensed, and the human forgot to set up the indicator, the computer would space paper through the printer with a speed that will amaze you, no, terrify you -- before you can get your finger on

the right button to stop it. The computer will search for the indicator until there is no paper left in the printer. The computer can not distinguish between right and wrong -- so the human instructing the computer must be cautious -- the human who is to receive the computer's output can and will tell you if it's wrong.

Instructions must be given the computer in a prescribed format. First, the operation code which tells the computer "what to do." (GIVE EXAMPLES). Then, we have the "N" code -- this is an extremely useful code -- for it has the ability to modify the instructions -- it can limit the number of characters involved in a move or compare operation, or indicate the results of a previous compare -- high, low, or equal, or it can indicate which input/output device is to be used. We will use the "N" code very much in our programming problem. The "A" address gives the memory location we are beginning with or working "from" -- the "B" address gives the location "to."

(SHOW TRANSPARENCY #7)

Problem: we have a Master Inventory Tape Record consisting of 4 character stock #, 5 character description, and 5 character quantity. The master record is to be up-dated by a transaction tape of the same format, and a new master tape is to be written in the same format. The master tape is in ascending sequence, transaction tape is in ascending sequence. The transaction tape contains receipts only and does not contain any records that do not match the master tape. Sequence of both records is stock #.

(SHOW TRANSPARENCIES #8 AND #9)

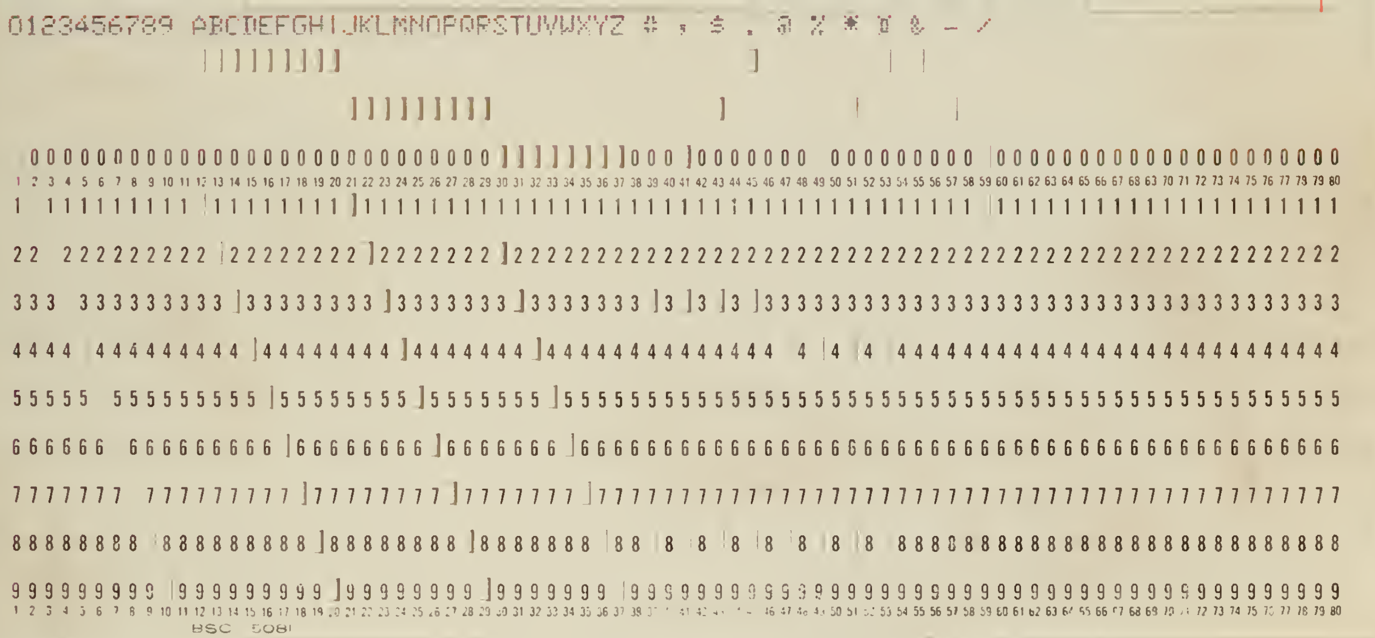
Explain the computers repetoire of instructions using instruction sheet handout. Code the problem on a large program record and have the students write along. Make sure each instruction is carefully explained as it is written.

Hand out solutions to problem.

SUMMARY:

Again I would like to emphasize that computers are the servants of human beings. They can do only what we tell them to do and their output can only be as accurate as the information given them. Are there any questions on this session?

CARD CODING SCHEME



HANDOUT

REMARKS

From Inst. Loc	HSM Location	OP	N	A		B		REMARKS	Box No.
		0	1	2	3	4	5		

Page Of

BLANK CODING SHEET

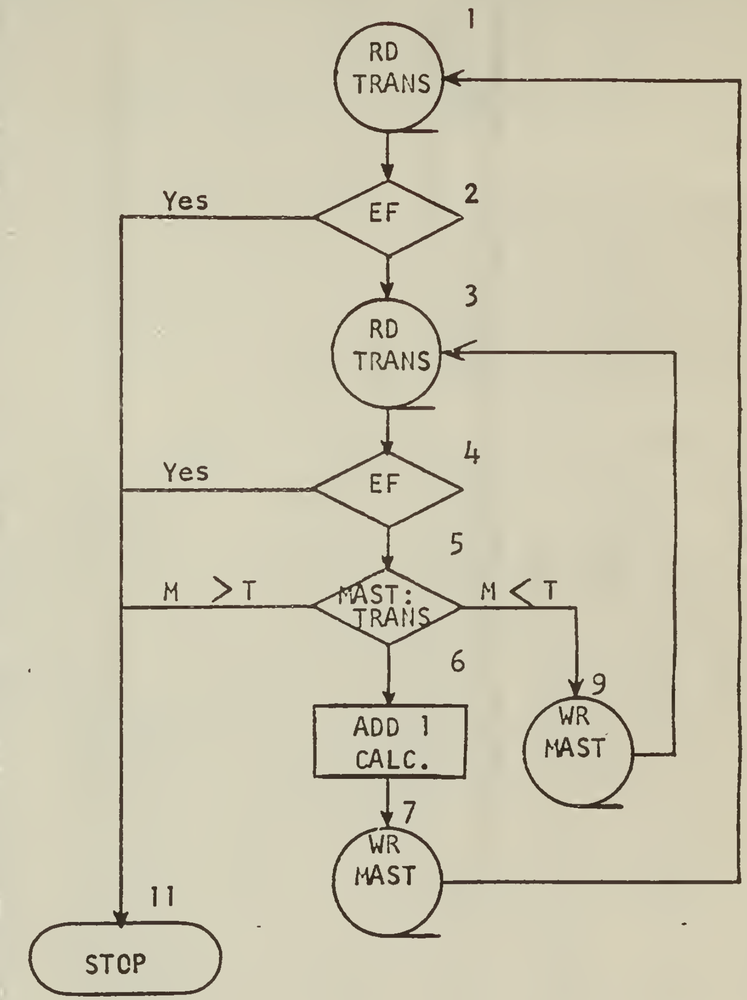
HANDOUT

CODER | G | Stock No. | Description | Quantity | G

REMARKS

From Inst. Loc	HSM Location	OP 0	N 1	A 2	A 3	B 4	B 5	Box No.
	Tran	RSV	14					
	Mast	RSV	14					
	Start 1	RD	2	Tran				1
		CTC	2	End,	Start 2			2
	Start 2	RD	3	Mast				3
		CTC	2	End,	Go			4
	Go	COM	4	Mast,	Tran			5
		CTC	1	End,	Next			5
		ADD	5	Mast,	Tran			6
		WR	4	Mast				7
		TC		Start	1			8
	Next	WR	4	Mast				9
		TC		Start	2			10
	End	HLT						11

REMARKS



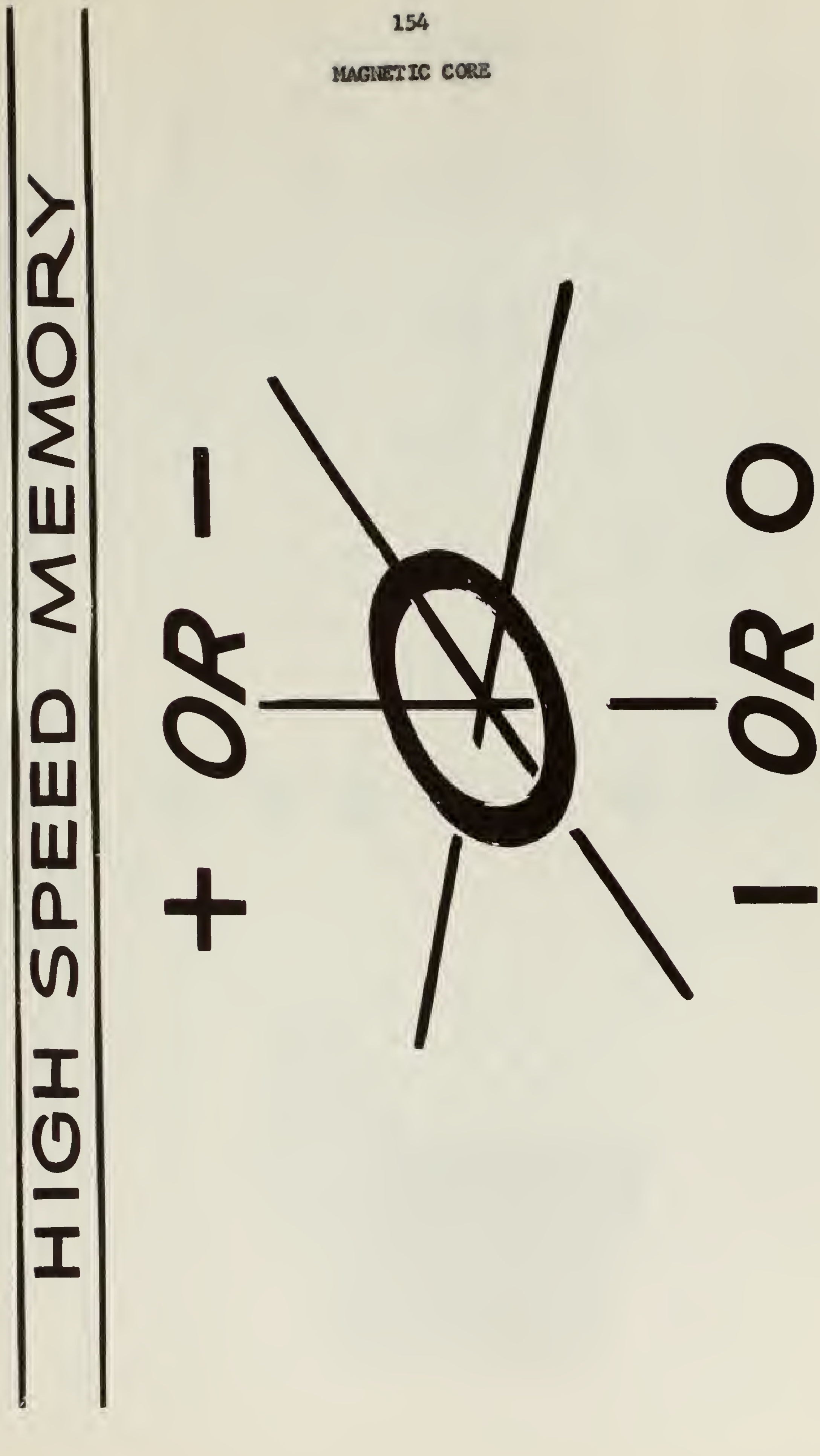
COMPLETED CODING SHEET

SUMMARY OF INSTRUCTIONS

Instruction	N	A	B	Remarks
Read	Device Number (Address)	HSM address of First character To be read	HSM address of last character To be read	Card Reader N-0 Paper Tape Reader N-1 Magnetic Tape 1-R N-2,3,4,5,6, or 7
Write	Device Number (Address)	HSM address of First character To be written	HSM address of Last character To be written	Card Punch N-0 Paper tape Punch N-1 Magnetic tape N-2,3,4,5,6, or 7 Printer L-R N-8
Move	Number of Characters To be moved Up to 35	HSM address of Left-most character to be Moved	HSM address of Destination	N-0 No move 1-9 up to nine characters A-Z to 35 characters L-R
Compare	Number of Characters To be compared up to 35	HSM address of First operand Left-most character to be Compared	HSM address of Second operand Left-most character to be Compared	B operand is subtracted from A operand starting at the left-most character terminating at Inequality of N-0 setting the PRI
Conditional Transfer of Control	N-0, none N-1, PRI N-2, EF/ED	HSM address of Next instruction if: N-1, PRI is + N-2, EF/ED present	HSM address of Next instruction if: N-1, PRI is - N-2, EF/ED not present	N-1 and PRI is zero, next instruction is executed
Transfer Control	Not req'd.	HSM address of next instruction to be executed	Not req'd.	Transfers "A" register to "P" Register
Halt	Not req'd.	HSM address of next instruction after start	Not req'd. Can be used For coding	Transfers "A" register to "P" Register if start is depressed
Add	Number of Characters In operands Include Sign position (up to 35)	HSM address of Sign position Of augend and Sum	HSM address of Sign Position Of addend	R-L sign of sum is algebraically correct
Subtract	Number of Characters In operands Include Sign position (up to 35)	HSM address of Sign position Of minuend and Difference	HSM address of Sign position Of subtrahend	R-L
Rewind	Device Number (Address) 2,3,4,5,6, or	Not req'd.	Not req'd.	Causes tape to completely rewind

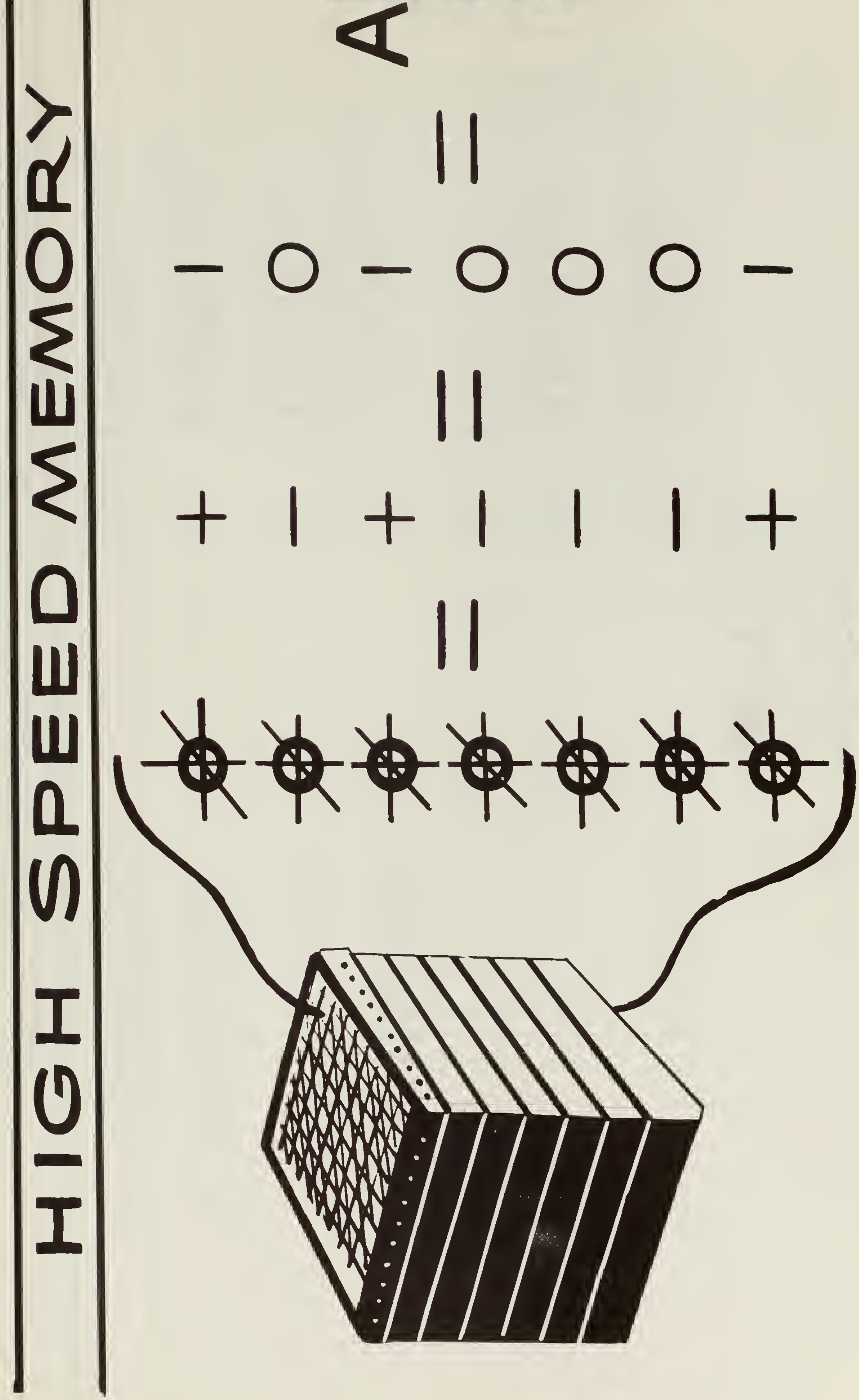
HANDOUT

MAGNETIC CORE



TRANSPARENCY 1

MAGNETIC CORE MATRIX



TRANSPARENCY 2

CODING

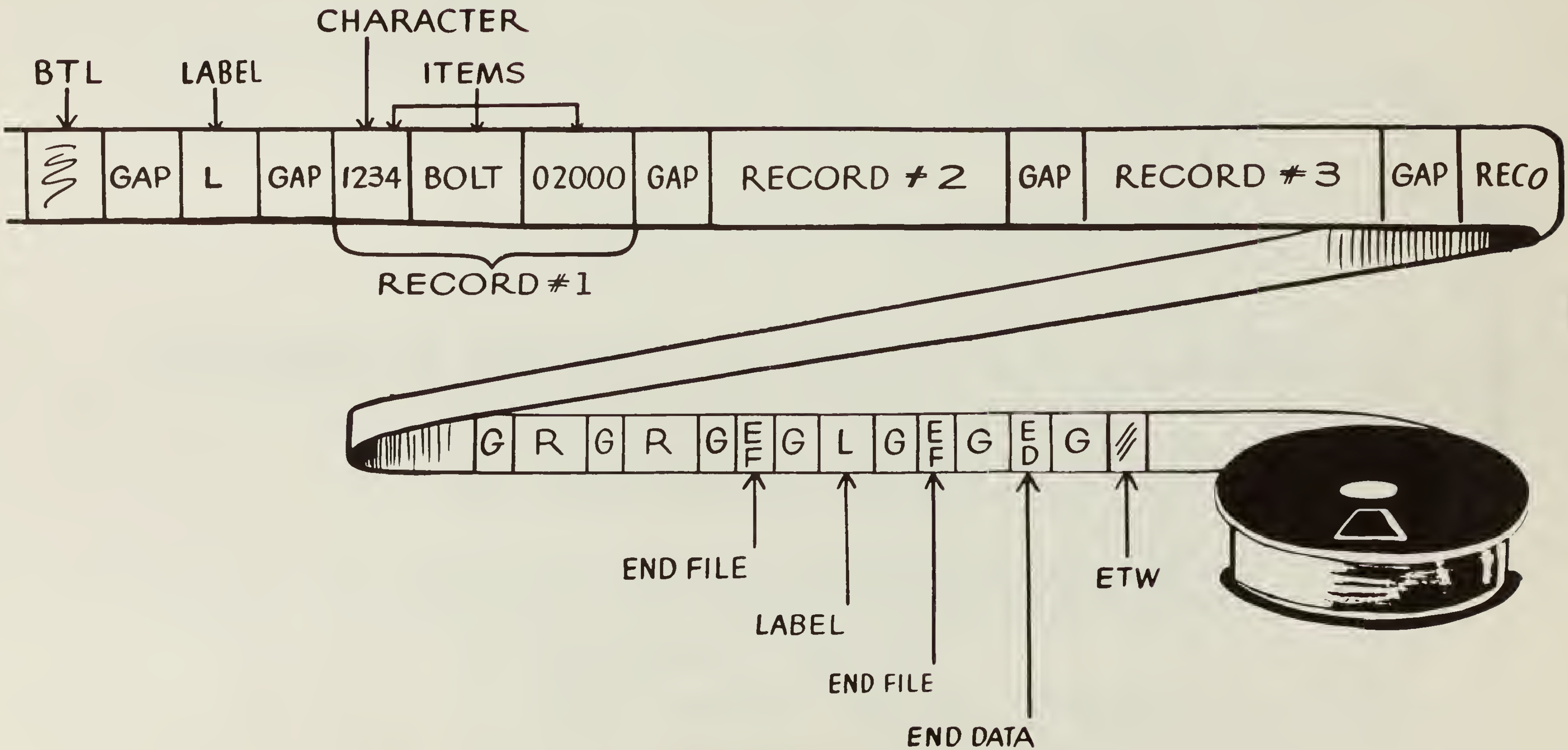
0 = 1000000	A = 1010001
1 = 0000001	B = 1010010
2 = 0000010	I = 0011001
3 = 1000011	J = 1100001
	R = 0101001
	S = 0110010
9 = 1001001	Z = 1111001

\+ 0011010

$ 1101011

TRANSPARENCY 3

ORGANIZATION OF DATA ON MAGNETIC TAPE



TRANSPARENCY 4

ORGANIZATION OF DATA ON MAGNETIC TAPE

ORGANIZATION OF DATA

ORGANIZATION OF DATA

CHARACTER = 0-9, A-Z, ETC

ITEM = A GROUP OF RELATED CHARACTERS

A QUANTITY.

AN EMPLOYEE NUMBER, ETC.

RECORD = A GROUP OF RELATED ITEMS.

A STOCK RECORD.

A PAYROLL RECORD, ETC.

FILE = A GROUP OF RELATED RECORDS.

A SUPPLY FILE

A PAYROLL FILE, ETC.

TRANSPARENCY 5

FIRST INSTRUCTION						2ND INSTRUCTION				
OPER CODE	N CODE	"A" ADDRESS		"B" ADDRESS		OPER CODE	N CODE	A ADDRESS		
		A_0	A_1	B_0	B_1			A_0	A_1	B_0

OPERATION CODE - WHAT IS TO BE DONE

"N" CODE OPERATION LIMIT OR I/O ADDRESS

"A" ADDRESS - MEMORY LOCATION "FROM

"B" ADDRES MEMORY LOCATION "TO"

TRANSPARENCY 6

Problem for a 100 position Hypothetical Computer

PROBLEM: Update of Master Stock Record File

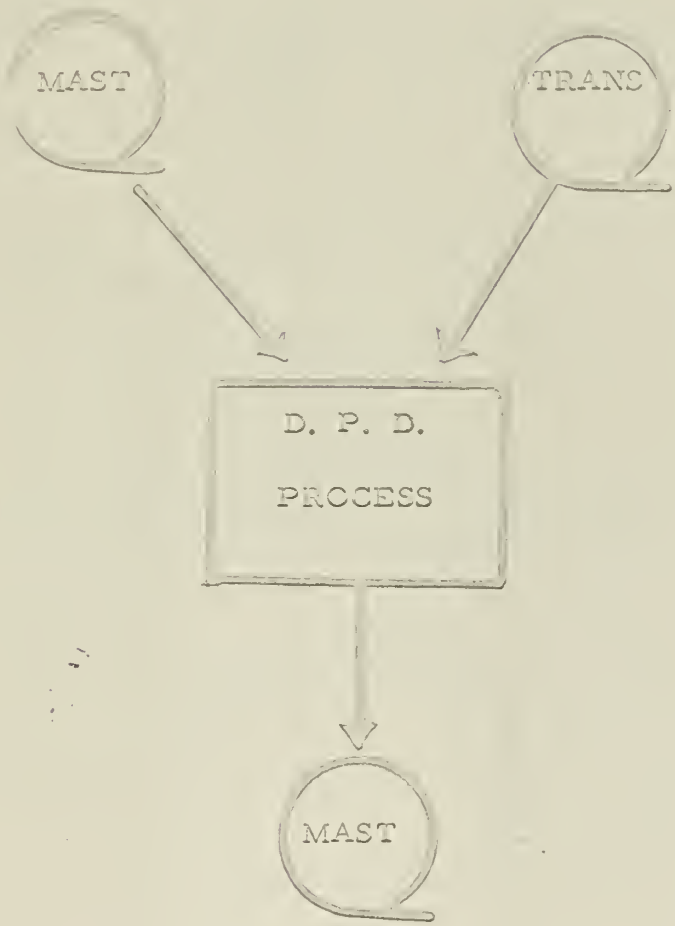
IN/PUT - Master File
Transaction File (receipts only)
Sequential ORDER
14 Characters per Record
1 Record per Block

4	5	5
stock no.	descript.	qty.

TRANSPARENCY 7

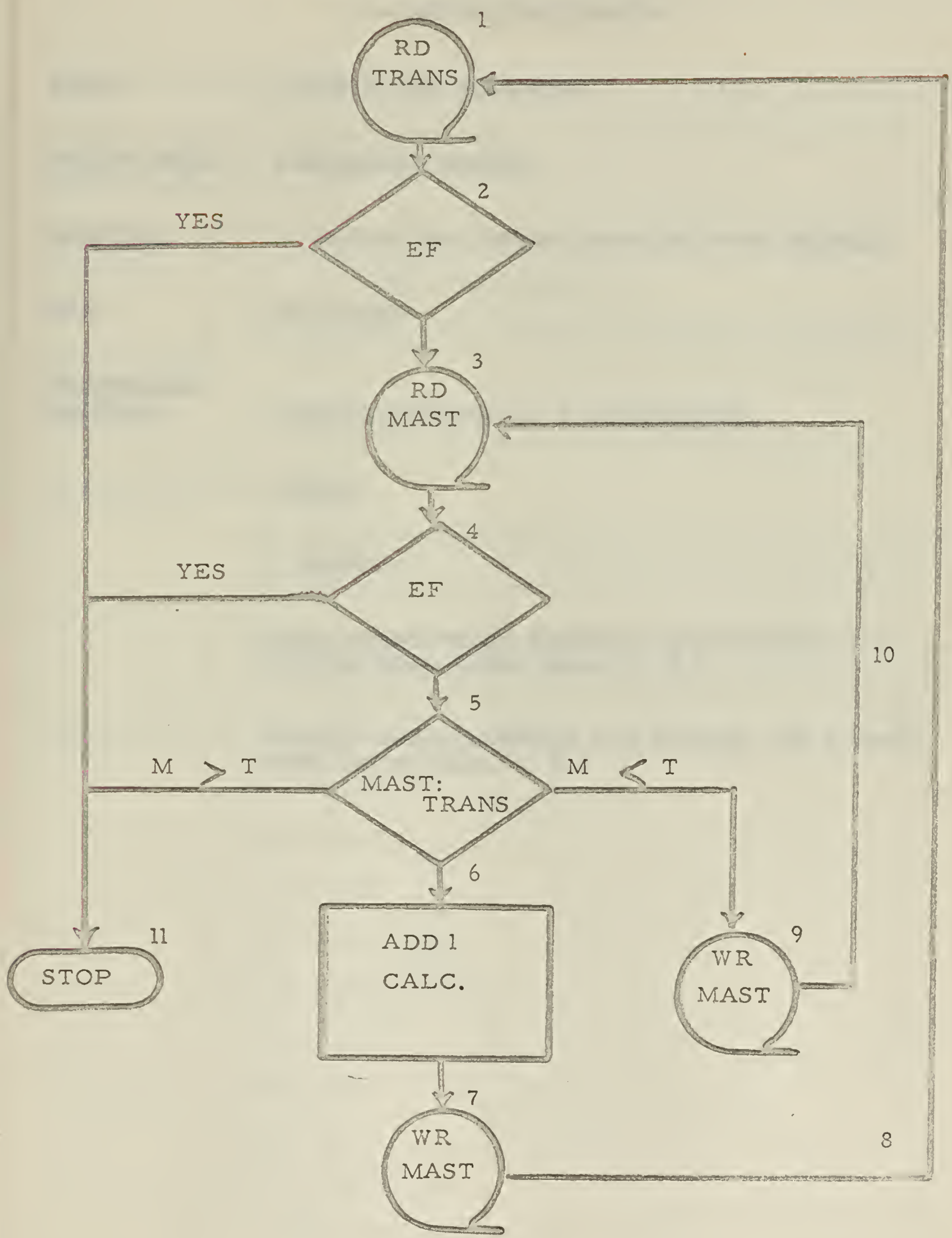
FLOW CHART OF PROBLEM

1. UPDATE MASTER TAPE FILE FROM TRANSACTION TAPE - WRITING OUT ON NEW TAPE



TRANSPARENCY 8

BLOCK DIAGRAM OF PROBLEM



TRANSPARENCY 9

G -- HYPOTHETICAL COMPUTER

COURSE: AUTOMATIC DATA PROCESSING

SESSION TITLE: HYPOTHETICAL COMPUTER

OBJECTIVE: To describe the internal operations of a computer.

TIME: 105 minutes

TRAINING AIDS REQUIRED: Overhead projector and 5 transparencies

Pointer

3 Handouts

Large program record (Available from Training Aids Library, MCAS, Cherry Point, N. C.)

Computer board (Available from Training Aids Library, MCAS, Cherry Point, N. E.)

HYPOTHETICAL COMPUTER

To describe the internal operations of a computer, we must know the language that the computer understands. This is known as the absolute machine language.

1. Let's review some of the terms used in the last session.

(SHOW TRANSPARENCY #1)

a. In our hypothetical computer we will use a 6 character instruction.

(1) The first character of the instruction is the "operation code."

(2) The next character is referred to as the "N" character.

(3) The next to characters are the "A" operand and,

(4) The next two characters are the "B" operand.

2. a. THE OPERATION CODE or command (1) character, tells the computer what operation to perform. Some examples are:

Read a card

Punch a card

Write a line

Read magnetic tape

Write magnetic tape

Compare A to B

Move A to B

Add A to B

Go to A if PRI set

b. The (PRI) (previous results indicator) is an indicator in the hardware which shows the results of a previous operation such as

positive, negative or Ø results after arithmetic operation or high, low or equal after a comparison operation.

c. The "N" character is used to limit the operation code, or to designate an input unit address, for example:

(1) In a move operation the "N" character can specify the number of characters to be moved.

(2) When using a read or write operation the "N" character would indicate the particular 1/0 unit to be activated. Example:

Tape station #4

Reader #1

Printer #1

(3) When testing for existing conditions the "N" character specifies the condition.

d. The A operand specifies the A address or memory location from. The B operand specifies the B address or memory location to. Example:

The contents of

Move	A	to	B
Compare	A	to	B
Add	A	to	B

or in the case of a transfer of control, the A and B operands will specify where the next instruction to be executed is located: Example: CTC 25, if condition met, if not, go to 46 (instruction located at).

e. Refer to previous handout of instructions to be used with the hypothetical computer.

During the last session you wrote a program using symbolic language, today we will use the absolute machine language.

Notice that the operation codes in absolute language are 1 character codes.

Absolute machine language is the only language that a computer can use as instructions. Symbolic languages must always be converted to absolute prior to being used by the machine.

f. We will be using the following instructions. (LIST INSTRUCTIONS. READ, WRITE, ADD, etc.)

Our master file is on magnetic tape. Notice that each record contains three (3) items.

1. A (4) character stock number.

2. A (5) character description.

3. A (5) character quantity.

For a total of fourteen (14) characters. This means that we must allocate 14 positions of our computer memory for each input area.

(SHOW TRANSPARENCY #3)

The master inventory tape file is to be up-dated by a transaction tape of the same format, and a new master tape is to be written in the same format.

The master tape and transaction tape are in ascending sequence stock number.

The transaction tape contains receipts only and does not contain any records that do not match the master tape.

Flow chart of problem:

Tapes 3 and 2 will be retained as history; this provides a method for reconstruction of the output should the need arise.

The "input" history is called father, grandfather, etc.

The "output" file is called son.

BLOCK DIAGRAM (SHOW TRANSPARENCY #4)

EXPLAIN EACH STEP IN THE DIAGRAM.

IF THERE ARE ANY QUESTIONS, CLARIFY.

II.

1. Introduce the hypothetical computer

a. Explain it's memory, size and addressing scheme. (00 thru 99)

b. Explain reason for * areas.

c. Explain that hand-written program would be converted to punched cards for actual loading into memory, or on a program of this small size we could key in the instructions from a keyboard.

d. Explain that larger computers like the 3301 use a 10 character address. This is due to the fact that 4 characters are required in each of the operands to take care of the capacity of the memory.
O N 0000 ZZ99.

2. Now let's write the program in absolute machine language.

a. Class participation.

b. Instructor use (LARGE PROGRAM RECORD) and (COMPUTER BOARD).

c. Set up data working areas first.

Our hypothetical computer has a memory of 100 positions. We have two inputs and each input contains 14 characters. Let's set aside memory

locations 99 through 86 for our transaction record input, and memory locations 85 through 72 for our master record input. * REMEMBER IN THE FILM DATA AND INSTRUCTIONS MUST BE KEPT SEPARATE IN COMPUTER MEMORY.

d. Write Program. Keep track of HSM locations. Leave A and B operands blank on transfer instructions until position in storage has been determined. Explain that this would be very difficult in an actual program. * MOVE PROGRAM INTO STORAGE AND EXPLAIN.

e. Give some advantages of using symbolic programming. (Symbolic operations, symbolic labels, storage allocation, loading.)

III.

1. Explain computer registers.

a. Registers control the sequence of events in a computer.

b. All computer are not alike in their register design and operation but they all accomplish the same thing.

c. There are always two phases required for each instruction cycle.

Access and execution

ACCESS -- During the access phase the instruction is read into the instruction registers.

EXECUTION -- After the instruction is read into the registers it is interpreted and the operation is executed. (UNVEIL RIGHT HAND SIDE OF MOBILE BOARD).

IV. The registers in our hypothetical computer are:

a. Arithmetic Unit -- Used for adding, subtracting, data multiplying and dividing are performed by repetitive additions or subtractions.

b. This unit is also used for making comparisons of alpha numeric characters and indicating the results in a PRI register (previous results indicator). For example, after an arithmetic operation the PRI would be +, - or 0 depending on the size of the numbers and the operation involved. Also after a comparison the PRI would be +, - or 0 depending on the characters in the comparison. The results would be referred to as high, low, or equal.

c. The P Register contains the address location of the next instruction to be executed. This register is incremented by 6 each time an instruction is accessed. It may also be reset on a transfer of control operation to the A or B register setting.

d. The "O" Register contains the operation code, and tells the machine what is to be done.

e. The "N" Register stores the "N" code which limits the operation or denotes the input/output unit address. (EXPLAIN)

f. The "A" and "B" Registers receive the address location of the A and B operands. The operands specify the locations of the data to be operated upon or the position in memory to which control is to be transferred. (EXPLAIN)

g. Memory Register is the medium by which the instructions or data are transferred to or from HSM.

h. Memory address register controls the access of instructions into the register by counting each character.

i. Bus-Adder will not be discussed in this problem.

j. Console. Is the display panel used by the operator to interrogate the status of the computer memory and the operations being

performed. The visual display of the registers are usually located on the console.

k. Input/Output "1/0." This is the channel over which data travels to and from various input/output devices.

l. Access Instruction No. 1 (ADDRESS REGISTERS SETTING AT 0)

1. Show flow into registers.

2. Explain operation of P register.

3. Explain operation of MAR register.

4. Explain that data would fill memory locations 86 through 99.

m. Access Instructions No. 5 and 6.

1. Explain as above.

2. Explain conditional transfer of control use of "P" register.

QUESTIONS

V. FINISHED PROBLEM

(HANDOUT #9)

This is a very elementary example of programming, used to show you the step by step method of solving a problem. From this simple program we can see some of the problems that systems analysts and programmers must overcome before a task can be given to the computer.

Writing the program is only half the job. After the program has been converted to an acceptable form (punched-cards) for the computer it is then checked out, first for validity or acceptance by the computer itself, and then in processing data on a trail run. After all checking and cor-

relations have been completed, the program must be documented. A complete record kept. This facilitates changes or reference at a later date, also run manuals and procedures must be prepared so that people will know how to operate with the program.

FIRST INSTRUCTION						2ND INSTRUCTION				
OPER CODE	N CODE	"A" ADDRESS		"B" ADDRESS		OPER CODE	N CODE	A ADDRESS		
		A_0	A_1	B_0	B_1			A_0	A_1	B_0

OPERATION CODE - WHAT IS TO BE DONE

"N" CODE OPERATION LIMIT OR ½ ADDRESS

"A" ADDRESS - MEMORY LOCATION "FROM

"B" ADDRES MEMORY LOCATION "TO"

TRANSPARENCY 1

INSTRUCTION FORMAT

Problem for a 100 position Hypothetical Computer

PROBLEM: Update of Master Stock Record File

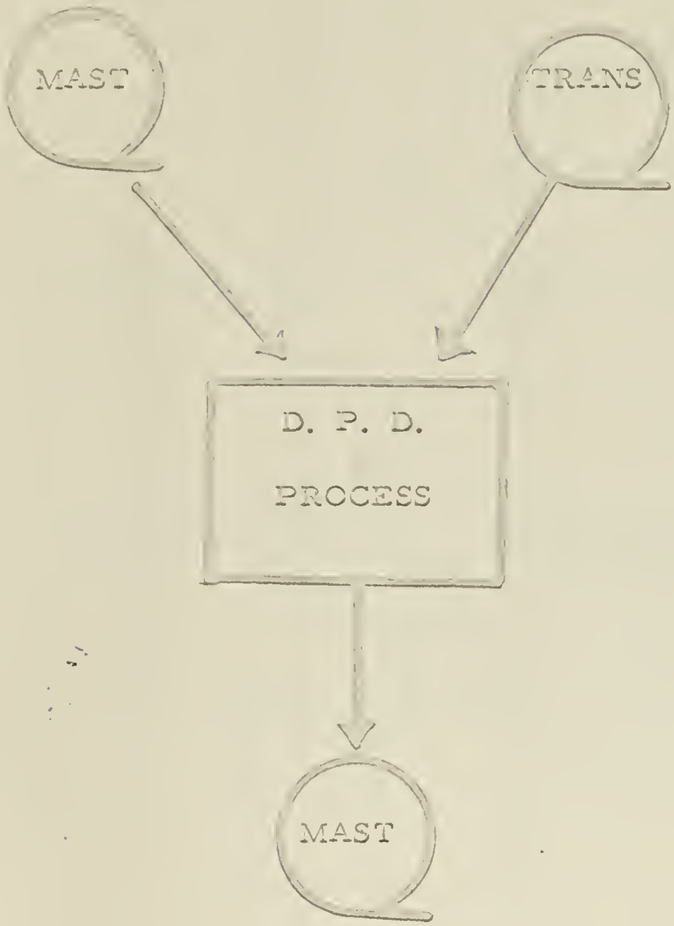
IN/PUT - Master File
Transaction File (receipts only)
Sequential ORDER
14 Characters per Record
1 Record per Block

4	5	5
stock no.	descript.	qty.

TRANSPARENCY 2

PROBLEM FOR HYPOTHETICAL COMPUTER

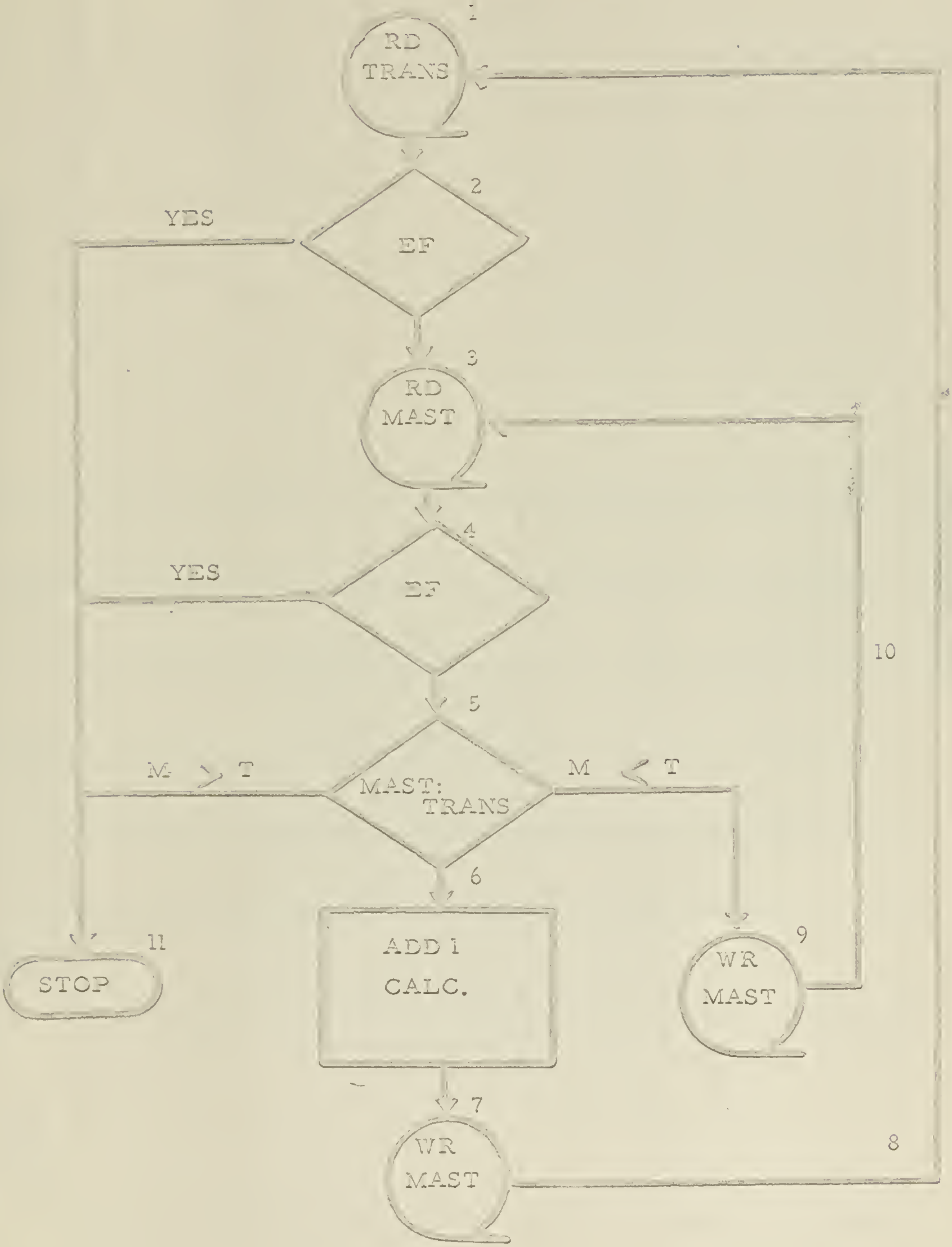
1. UPDATE MASTER TAPE FILE FROM TRANSACTION TAPE -WRITING OUT ON NEW TAPE



TRANSPARENCY 3

FLOW CHART OF PROBLEM

BLOCK DIAGRAM OF PROBLEM



TRANSPARENCY 4

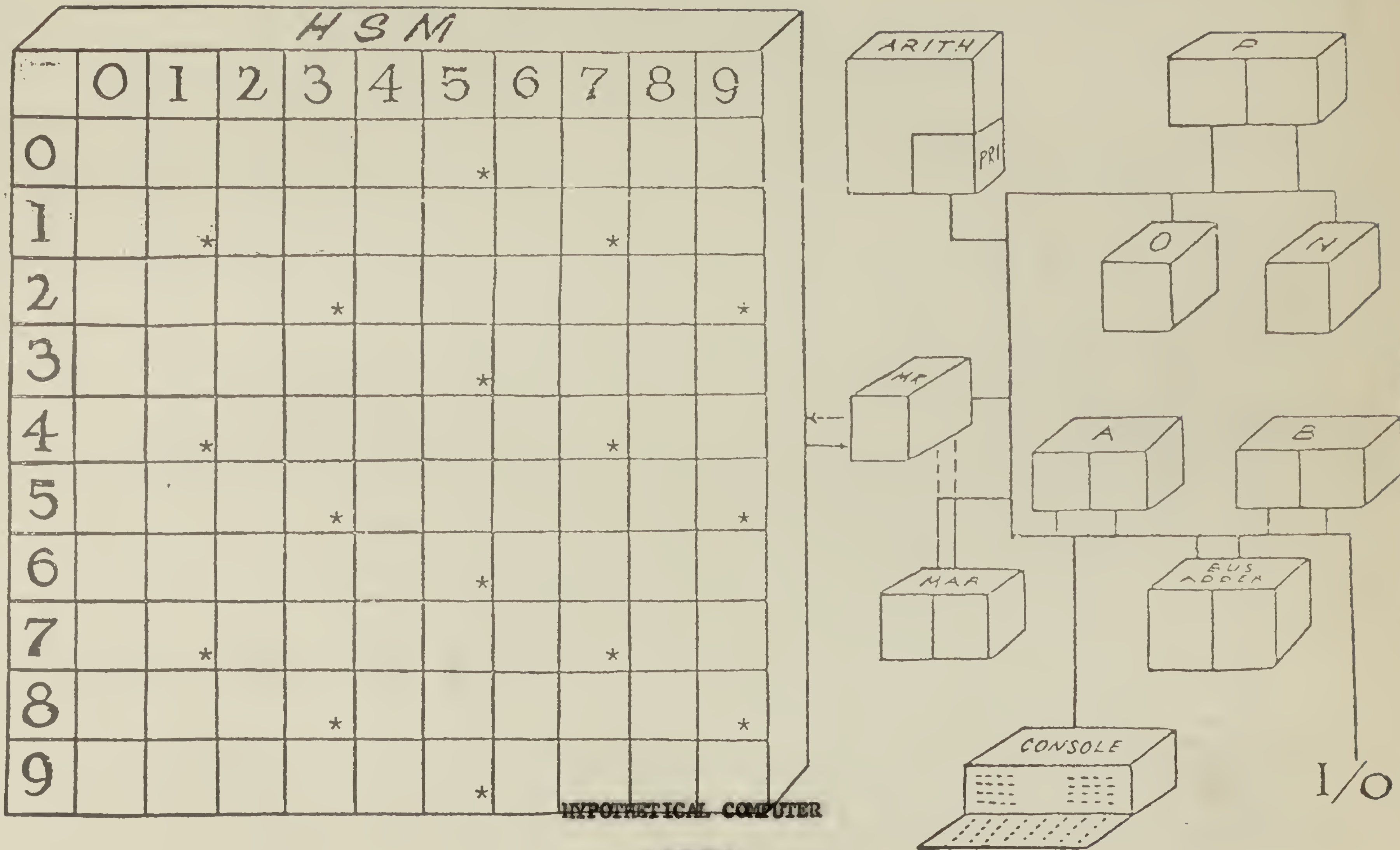
TITLE
CODER
REMARKS

DATE

From Inst. Loc	HSM Location	OP	N	A		B		REMARKS	Box No.
		0	1	2	3	4	5		

Page Of

BLANK CODING SHEET



HYPOTHETICAL COMPUTER

HANDOUT

Date May 15, 1964

TITLE Demo
CODER V. W. Seitz
REMARKS

From Inst. Loc	HSM Location	MACHINE CODE OP 0	N 1	A 2	A 3	B 4	B 5	REMARKS	Box No
									1
48	00	R	2	8	6	9	9	Read Transaction	2
	06	T	2	6	6	1	2	Test for EF/ED	3
60	12	R	3	7	2	8	5	Read Master	4
	18	T	2	6	6	2	4	Test for EF/ED	5
	24	C	4	7	2	8	6	Compare Master to Transaction	5
	30	T	1	6	6	5	4	CTC	6
	36	+	5	8	5	9	9	Balance + Quantity	7
	42	W	4	7	2	8	5	Write Master (Hit)	8
	48	I	-	0	0	-	-	TC to Start	9
	54	W	4	7	2	8	5	Write Master (no-hit)	10
	60	I	-	1	2	-	-	TC to Read New Master	
	66	H	-	-	-	-	-		

	0	1	2	3	4	5	6	7	8	9
0	R	2	8	6	9	9	T	2	6	6
1	1	2	R	3	7	2	8	5	T	2
2	6	6	2	4	C	4	7	2	8	6
3	T	1	6	6	5	4	+	5	8	5
4	9	9	W	4	7	2	8	5	I	-
5	0	0	-	-	W	4	7	2	8	5
6	I	-	1	2	-	-	H	-	-	-
7	-	-	M	A	S	T	E	R	→	
8	→						T	R	A	N
9	S	A	C	T	I	O	N	→		

COMPLETED CODING SHEET

CHAPTER V

SUMMARY AND CONCLUSION

Necessity for Management Training

Computer usage has been expanding very rapidly in the last few years. This expansion will be the cause of numerous future changes in management operations. A survey of McKinsey and Company supports this theory and points out that the computer is not merely a machine but a new resource to be utilized in solving long standing management problems. To utilize this resource effectively will require a basic change in approach. Imagination in its use by some managers paid off handsomely.

To accomplish this realization management must become involved by exerting executive leadership and management controls of the system. They must therefore be trained in the methods, procedures and capabilities of data processing if they are to exploit computer systems properly.

The Marine Corps and Computers

The Marine Corps, currently in an expansionary phase in the acquisition of computers, utilizes them primarily in Supply Accounting, Personnel Accounting, Fiscal and Disbursing, and other logistical support programs. However, there is no segment of Marine Corps activity which is not directly influenced by the use of electronic computers.

The ever expanding complexity of all computer oriented systems demands that all officers have an understanding of automatic data processing. This need has been recognized but the implementation has been slow. The service academies, Amphibious Warfare School, and the Command and Staff College are providing training, but the number of students is limited.

A step in the right direction towards orientation of all officers is an internal course conducted at Headquarters Marine Corps. No detailed lesson plans for that course are presently available. Concern for the need of a semi-standard course to be presented locally at all installations has been expressed by certain Marine Corps officials.

The Cherry Point Approach

An approach to training of management personnel was conducted at the Marine Corps Air Station, Cherry Point, N. C. A total of three hundred fifty-eight management personnel at Cherry Point have attended an orientation course. This course covered data processing in general coupled with specific reference in the latter portion to the computer system at Cherry Point and the utilization of the system. It was originally thirty hours in length, but was reduced to twenty-three hours due to certain aspects of management resistance.

An analysis of critique sheets on the course overwhelmingly indicates that the course is logically organized, well presented, and of assistance to managers in accomplishment of their jobs.

<u>Conclusions</u>:

1. That training or orientation of all Marine Corps Officers in the "mystic" of the computer is of primary importance to its efficient and economical use.

2. That a semi-standard course of two main segments be developed for use at all Marine Corps installations. The first section of the course should cover data processing in general and be standardized. A possible design for this portion is presented in Chapter IV. The second section of the course should cover hardware, software, and systems application. This should be non-standard and developed by each command to fit its needs.

3. That the course not be over twenty-four hours in length.

4. That the area of specialized training which has not been touched upon in the presentation, is a fertile subject for further investigation.

BIBLIOGRAPHY

Public Documents

U. S. Bureau of the Budget. Automatic Data Processing Glossary. Washington, D. C.: Government Printing Office, 1962.

U. S. Comptroller General. Review of Problems Relating to Management and Administration of Electronic Data Processing Systems in the Federal Government. Report B-115369, April 30, 1964.

U. S. House of Representatives. Committee on Post Office and Civil Service. Use of Electronic Data Processing Equipment in the Federal Government. Report No. 858, 88th Cong., 1st Sess., 1963.

U. S. Senate. Committee on Government Operations. Report to the President on the Management of Data Processing. February, 1965.

Books

Albers, Henry H. Organized Executive Action. New York: John Wiley & Sons, Inc., 1962.

Baumes, Carl G. and Pflomm, Norman E. Management's Role in Electronic Data Processing. New York: National Industrial Conference Board, 1959.

Bell, William D. A Management Guide to Electronic Computers. New York: McGraw-Hill Book Co., Inc., 1957.

Brozen, Yale. Automation: The Impact of Technological Change. Washington, D. C.: American Enterprise Institute, 1963.

Canning, Richard G. Electronic Data Processing for Business and Industry. New York: John Wiley & Sons, Inc., 1956.

Gallagher, James D. Management Information Systems and the Computer. New York: American Management Association, 1961.

Gregory, Robert H. and Van Horn, Richard. Automatic Data Processing Systems. Belmont, California: Wadsworth Publishing, Inc., 1964.

Laden, Hyman N. and Gildersleeve, T. R. Systems Design for Computer Applications. New York: John Wiley and Sons, Inc., 1963.

Marting, Elizabeth (ed.). Establishing an Integrated Data Processing System. New York: American Management Association, 1956.

Phillipson, Morris (ed.). Automation: Implications for the Future. New York: Random House, Inc., 1962.

Articles and Periodicals

Bekker, John A. "Automation, Its Impact on Management," Advanced Management, (December, 1959), pp. 20-24.

Berkeley, Edmund C. "Education in the Field of Computers and Data Processing," Computers and Automation, (July, 1965).

Brandon, Dick H. "The Computer Personnel Revolution," Advanced Management, (December, 1959), pp. 22-23.

________. "The New Breed of Software," Computers and Automation, (April, 1965), p. 17.

________. "Editorial on Management Education," Computers and Automation, (December, 1965), p. 39.

Campbell, Sullivan G. "Time Sharing Some Problems, Potentials and Implication," Data Processing Magazine, (September, 1965), pp. 30-33.

"Data Transmission Systems," Datamation, (December, 1965), pp. 51-53.

Diebold, John. "Urgent Need: All Out Automation," Nation's Business, (July, 1961), pp. 29-31, 58-59.

________. "The Still Sleeping Giant," Harvard Business Review, (September-October, 1964).

Galler, Bernard A. "Software Trends," Data Processing Magazine (March, 1965), pp. 56-57.

Garrity, John T. "Top Management and Computer Profits," Harvard Business Review (July-August, 1964), pp. 6-12, 172-174.

Gill, William A. "Sound Management and Effective Use of Computers in the Federal Government," Computers and Automation (April, 1965), pp. 14-17.

Greene, General Wallace M. Jr., Commandant of the Marine Corps. "Defense Top Management Annual Review," Armed Forces Management, (November, 1964).

________. "Defense Top Management Annual Review," Armed Forces Management, (November, 1965).

Haga, E. J. "Education in Computers and Data Processing," Computers and Automation, (November, 1965), p. 11.

Investment Bankers Association of America. "A Survey and Study of the Computer Field," Computers and Automation, (January, 1963), pp. 15-25.

Giesler, Murray A. and Steger, Williber A. "How to Plan for Management in New Systems," Harvard Business Review, (September-October, 1963), pp. 103-110.

Janinski, Frank J. "Adapting Organization to New Technology," Harvard Business Review, (January-February, 1959), pp. 79-86.

Lipstreu, Otis. "Organizational Implications of Automation," Journal of the Academy of Management, (August, 1960), pp. 119-120.

Magnis, N. E. "Management and the EDP Installation," Data Processing Magazine, (March, 1965), pp. 48-51.

McCrohan, James. "Government Sets Data Processing Guidelines," Datamation, (April, 1965), pp. 70-71.

Silberman, Charles E. "The Real News About Automation," Computers and Automation, (January, 1965), pp. 124-126, 220-222, 227-228.

Snyder, Robert G. "Programming Documentation," Datamation (October, 1965), pp. 44-48.

Trimble, George R. Jr. "Using a Computer to Simulate A Computer," Data Processing Magazine, (October, 1965), pp. 18-23.

"The $5 Billion World Market for Computers," Business Week, (February 19, 1966) pp. 110-132.

"User Oriented Computer Systems: Some Implications for Public Administration," Data Processing Magazine, (February, 1965), pp. 16-18.

Manuals and Pamphlets

Application Documentation Guide. Camden, N. J.: The Radio Corporation of America, 1963.

An Approach to the Basic Techniques of Systems Analysis. Camden, N. J.: The Radio Corporation of America, 1961.

Data Processing Systems. Washington, D. C.: Bureau of Aeronautics, 1957.

Flow Charting and Block Diagramming Techniques. White Plains, N. Y.: International Business Machines Corporation, 1961.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

General Information Manual: An Introduction to IBM Punched Card Data Processing. White Plains, N. Y.: International Business Machines Corporation, 1960.

General Information Manual: Introduction to IBM Data Processing Systems. White Plains, N. Y.: International Business Machines Corporation, 1960.

Murphy, E. A. Jr. _Do You Talk Computerese_. Philadelphia, Penna.: Honeywell Corporation, 1960.

N. B. C.'s Election Night Management Information System. Camden, N. J.: The Radio Corporation of America, 1964.

Reference Manual: IBM Magnetic Tape Units. White Plains, N. Y.: International Business Machines Corporation, 1961.

Reference Manual: IBM Study Organization Plan and Documentation Techniques. White Plains, N. Y.: International Business Machines Corporation, 1961.

Seitz, Vernon W. _Computer Concepts for Management_. New York: The Radio Corporation of America, 1964.

Ware, William E. _Numbering Systems_. Pottstown, Penna.: Honeywell Corporation, 1962.

What Every Businessman Should Know About Electronic Brains. New York: Sperry Rand Corporation, 1961.

Reports

Garrity, John T. _Getting the Most Out of Your Computer_. A Report of A Survey of Company Approaches and Results Prepared by McKinsey and Company. New York: McKinsey and Co., 1963.

U. S. Marine Corps. _Report Submitted to the Office of Management Information_. Prepared by the Data Systems Division, Headquarters, Marine Corps, Washington, D. C., 1966.

Unpublished Material

Gill, William A. "Economic Considerations in the Use of Electronic Computers." An Address before the American Management Association during A briefing session on Computer Economics. New York: June 29, 1962.

Harris, Jack W. and Hieber, George A. "Introduction to Automatic Data Processing and UDAPS for IMAS." Cherry Point, N. C., 1964. (Mimeographed.)

Hill, Lieutenant Colonel Arthur T. "Marine Cor;s Position and Philosophy on Computer Education," A Seminar of Military Computer Education and Computer Center Directors, June 14 to June 17, 1965. West Point: 1965. (Typewritten.)

Sarnoff, David. "The Social Impact of Computers." An Address before the American Bankers Association National Automation Conference. New York: July 16, 1964.

U. S. Marine Corps. "Air Bases Order 1500.8" Marine Corps Air Bases Eastern Area, Cherry Point, N. C., October 14, 1964.

U. S. Marine Corps. "Air Station Bulletin 1500, Marine Corps Air Station, Cherry Point, N. C., October 29, 1964.

U. S. Marine Corps. "Syllabus for Headquarters Marine Corps Automatic Data Processing Orientation Course." Headquarters Marine Corps, Washington, D. C., February 1966.

Other Sources

Personal interview with William W. Waters, Deputy Data Processing Officer, Marine Corps Air Station Cherry Point, N. C., at Washington, D. C., February, 1966.

Personal interview with Lieutenant Colonel Arthur T. Hill, Executive Officer Data Systems Division, Headquarters Marine Corps, Washington, D. C., February 17, 1966.

Personal interviews with Colonel Charles H. Green, Director, and Arthur T. Hill, Executive Officer Data Systems Division, Headquarters Marine, Corps, Washington, D. C. February 17, 1966.

thesH52726
Data processing training for Marine Corp
3 2768 002 05972 7
DUDLEY KNOX LIBRARY